2018

ΣΥΣΤΗΜΑΤΑ ΔΙΑΧΕΙΡΙΣΗΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ

Εργασια 2018-2019

|  |  |
| --- | --- |
| π16036 | ιωαννιδησ παναγιωτησ |
| π16097 | νικασ Διονυσης |
| π16112 | παραβαντησ αθανασιοσ |

Περιεχόμενα

[**1** **Μέρος 1ο** 2](#_Toc535958066)

[1.1 Εισαγωγή 2](#_Toc535958067)

[1.2 Ερώτημα A 3](#_Toc535958068)

[1.2 Ερώτημα B 8](#_Toc535958069)

[1.3 Ερώτημα Γ 15](#_Toc535958070)

[1.4 Ερώτημα Δ 22](#_Toc535958071)

[1.5 Ερώτημα Ε 27](#_Toc535958072)

[**2** **Μέρος 2** 28](#_Toc535958073)

[2.1 Ερώτημα Α 28](#_Toc535958074)

[2.2 Ερώτημα Β 38](#_Toc535958075)

[2.3 Ερώτημα C 39](#_Toc535958076)

# **Μέρος 1ο**

## Εισαγωγή

Το 1ο ερώτημα της εργασίας προϋποθέτει τη δημιουργία μιας βάσης δεδομένων που θα περιέχει όλα τα στοιχεία των δυο CSV αρχείων που μας δίνονται. Για την σωστή ακολουθία ενεργειών, έχουμε συμπεριλάβει το αρχείο **create\_table.sql** που βρίσκεται στο φάκελο **part\_1**.

Το αρχείο είναι χωρισμένο σε κατηγορίες που περιέχον τις εντολές για τις παρακάτω ενέργειες:

* Δημιουργία του πίνακα **accident\_information**.
* Αντιγραφή των δεδομένων του αρχείου **db2\_Accident\_Information.csv** στον πίνακα **accident\_information**.
* Δημιουργία του πίνακα **vehicle\_information**.
* Αντιγραφή των δεδομένων του αρχείου **db2\_Vehicle\_Information.csv** στον πίνακα **vehicle\_information**.

Τα σχετικά COPY queries δεν περιλαμβάνουν έγκυρα path για τα .csv αρχεία, επομένως είναι αναγκαία η αντικατάστασή τους με τα σωστά path για το εκάστοτε προγραμματιστικό περιβάλλον.

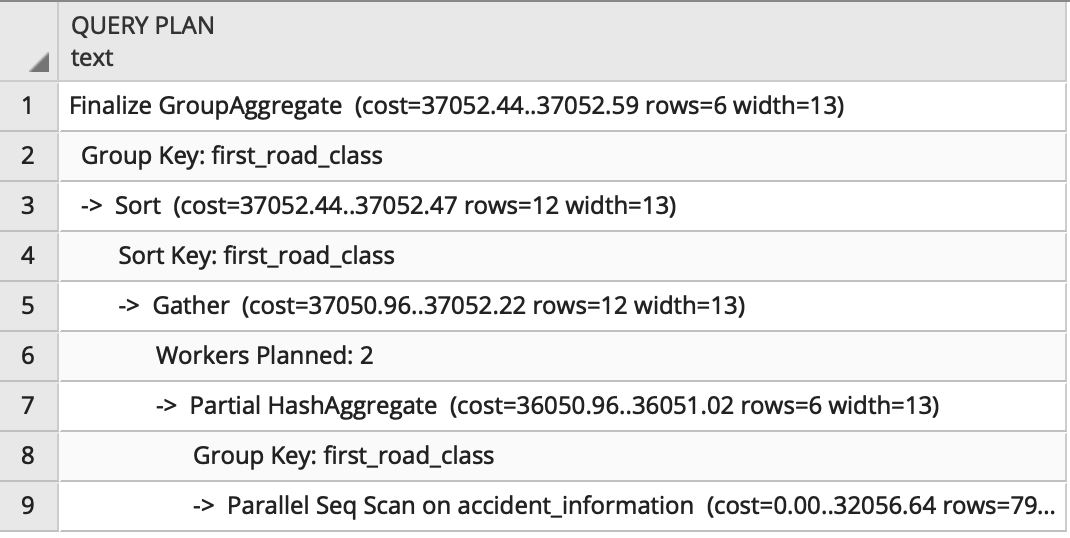
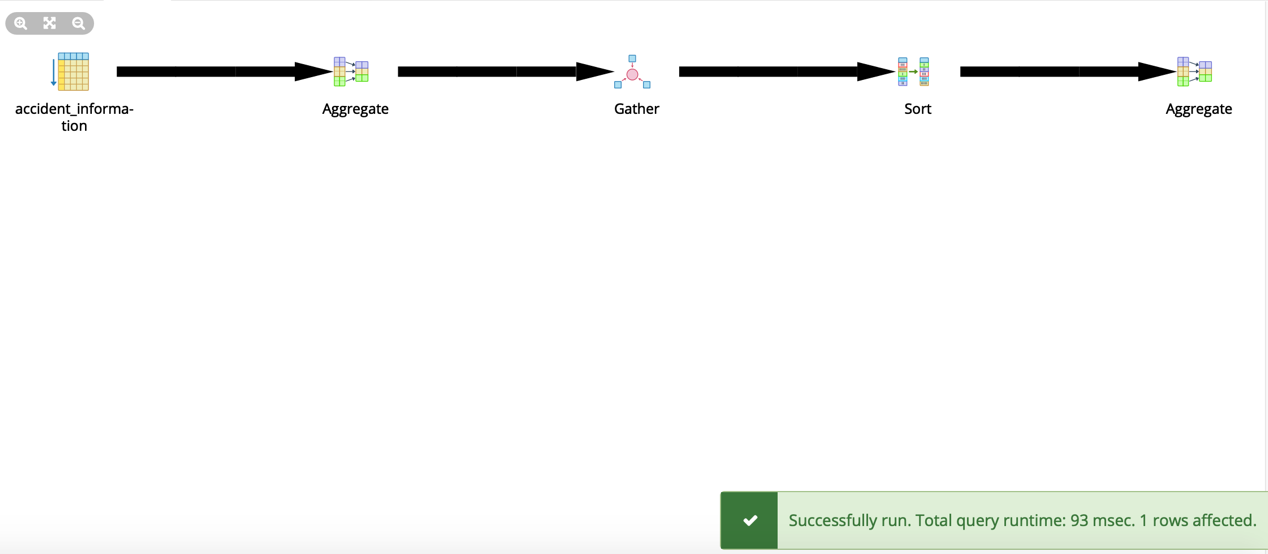
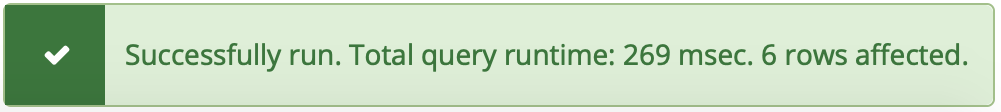
Οι απαντήσεις για κάθε ερώτημα του 1ου μέρους βρίσκονται στους φακέλους A\_queries, B\_shared\_buffers, C\_parallel\_query, D\_indexes και E\_partitions αντίστοιχα.

## Ερώτημα A

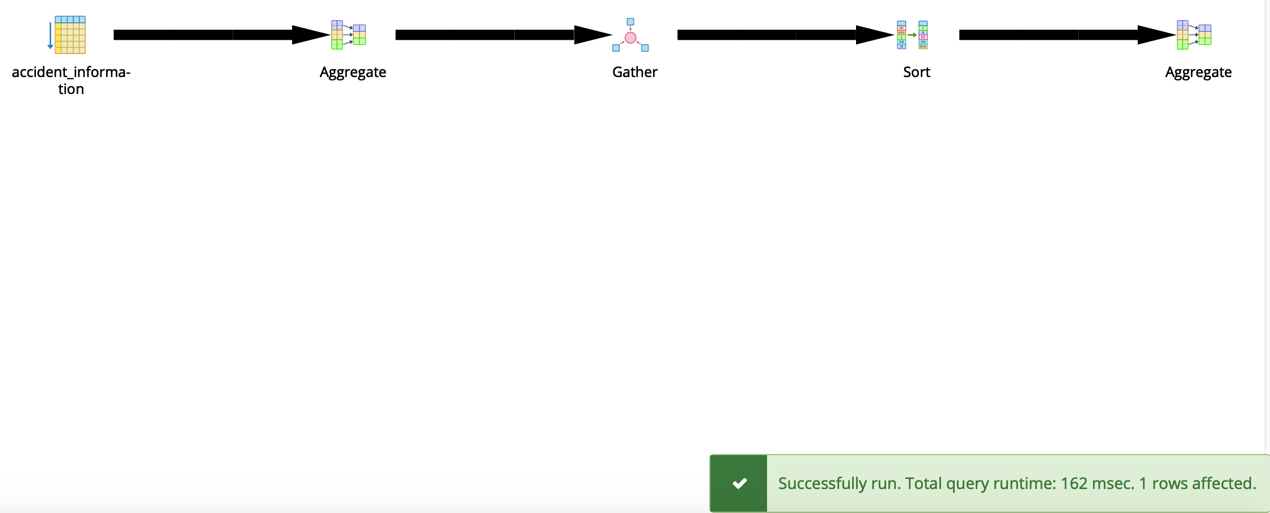
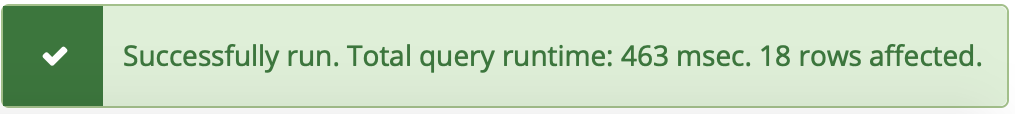
Στο πρώτο ερώτημα δημιουργούνται τα queries που θα χρησιμοποιηθούν κατά τη διάρκεια του πρώτου μέρους της εργασίας. Πριν την εκτέλεσή τους εκτελούμε την εντολή VACUMM FULL όπως ζητήθηκε για να κάνουν refresh τα στατιστικά των πινάκων. Στη συνέχεια θα εκτελέσουμε ένα ένα με τη σειρά τα queries χρησιμοποιώντας τις προεπιλεγμένες ρυθμίσεις της Postgres και χωρίς να έχουμε δημιουργήσει ευρετήρια.

Έτσι έχουμε για κάθε query πρώτα το χρόνο εκτέλεσης, μετά την ανάλυση με EXPLAIN και τέλος το QUERY PLAN :

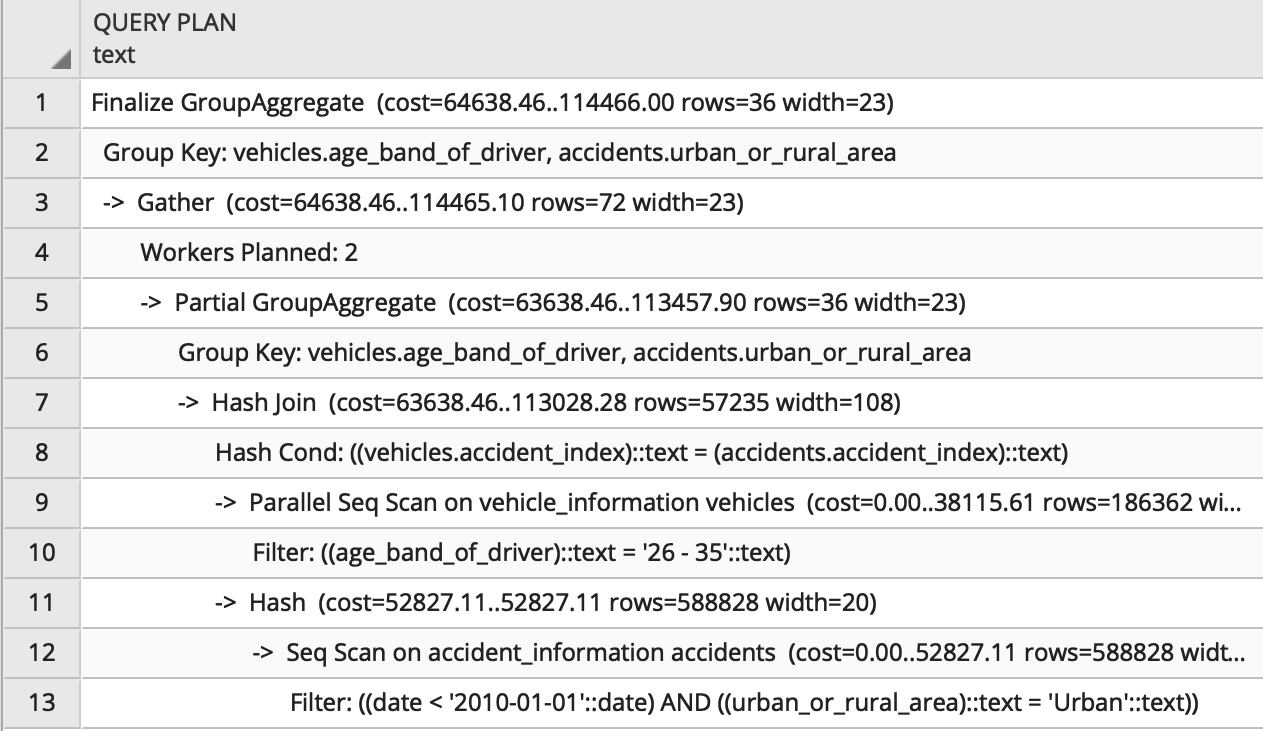
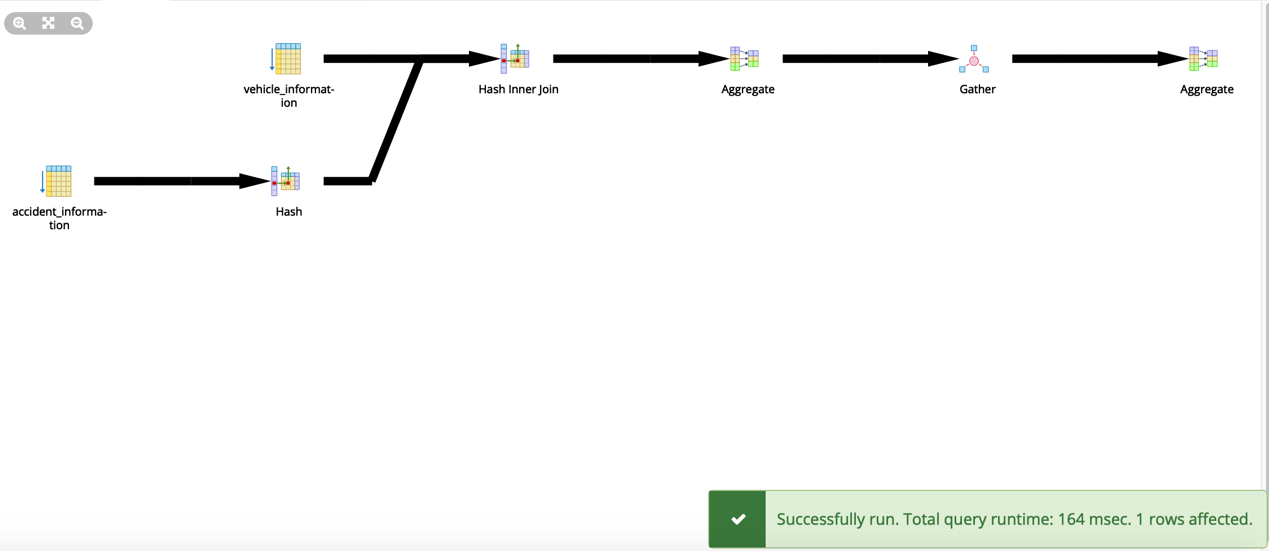
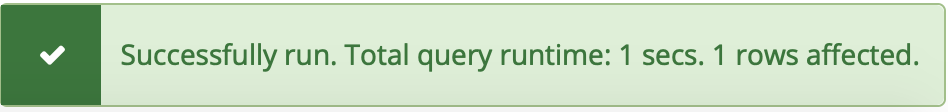
1. Πόσα ατυχήματα έγιναν ανά κατηγορία δρόμου (road class)



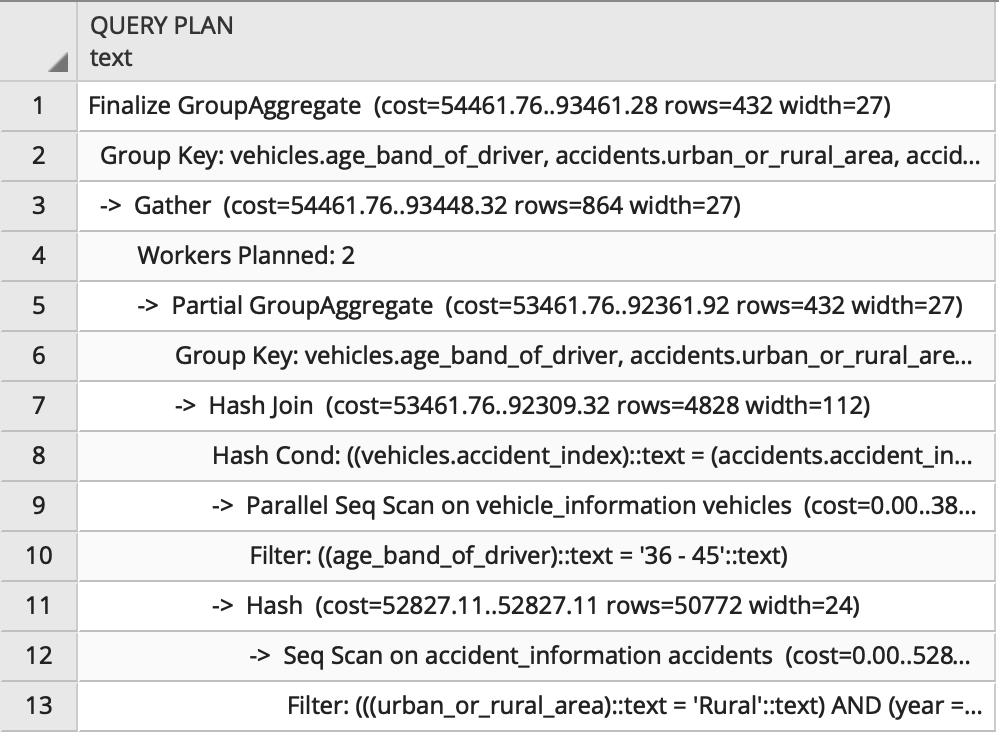
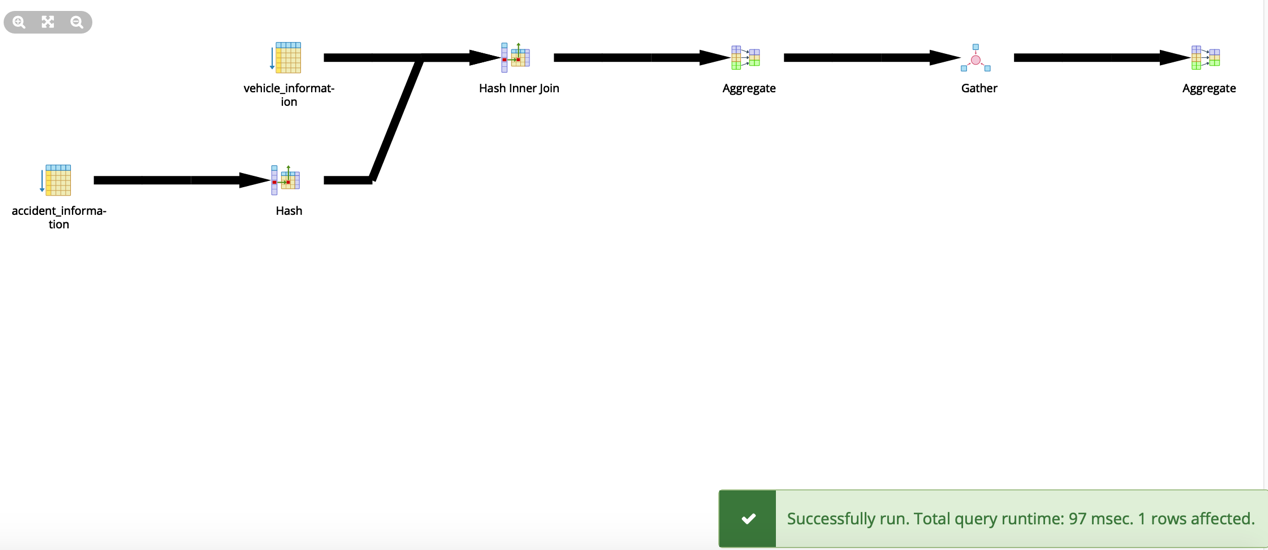
1. Πόσα είναι τα ατυχήματα ανά κατηγορία δρόμου και ανά κατηγορία ατυχήματος (Accident Severity).



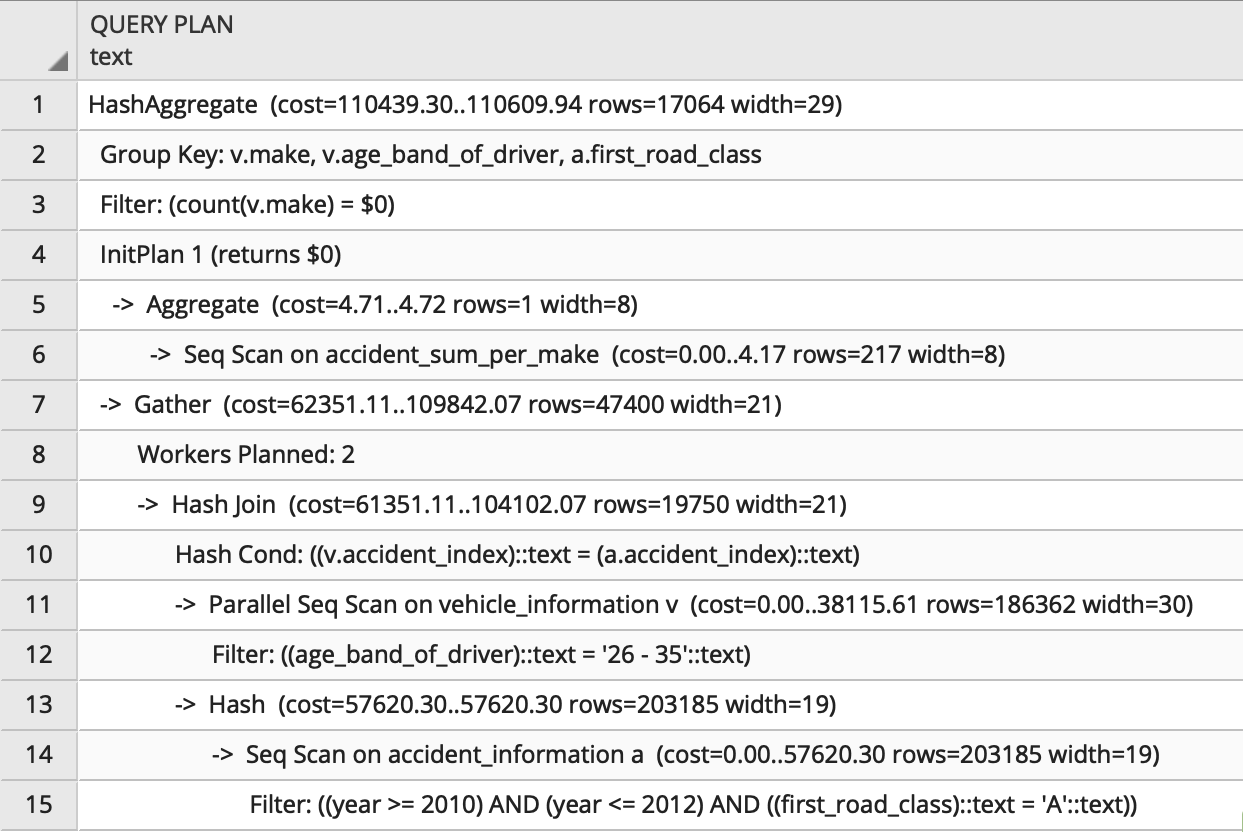
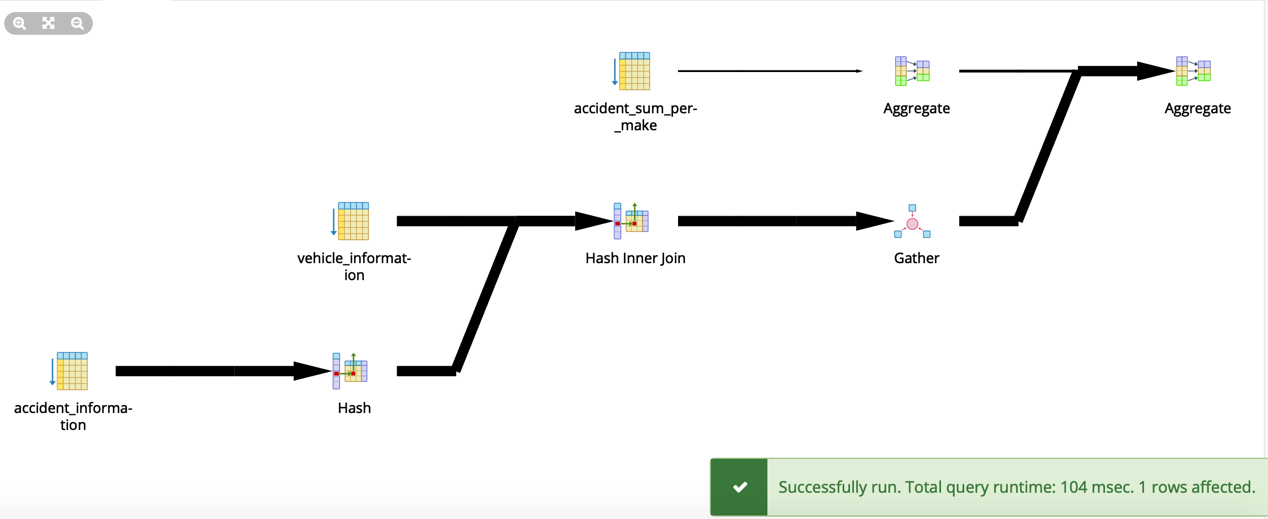
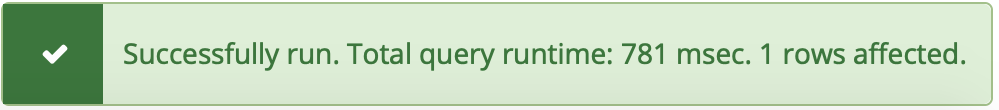
1. Πόσα ατυχήματα έγιναν σε αστική περιοχή πριν από την 1/1/2010 και η ηλικιακή κατηγορία του οδηγού είναι 26-35.



1. Πόσα ατυχήματα έγιναν σε αγροτική περιοχή το έτος 2012 και η ηλικιακή κατηγορία του οδηγού είναι 36-45.

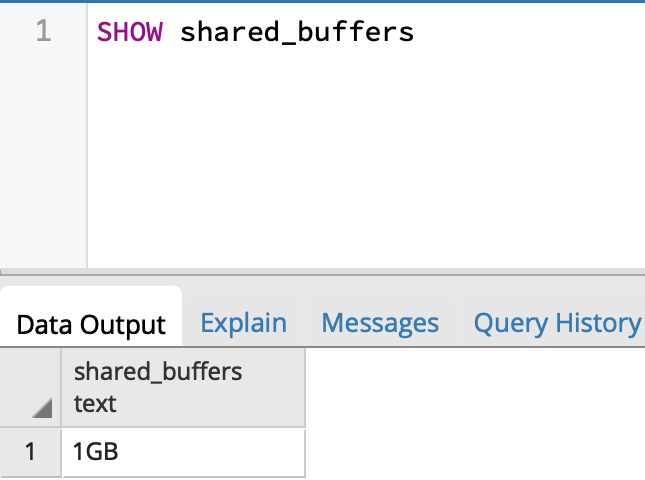


1. Ποιος είναι ο κατασκευαστής του οχήματος που έχει τα περισσότερα ατυχήματα μεταξύ των ετών 2010 και 2012, η ηλικιακή κατηγορία των οδηγών είναι 26- 35 και η κατηγορία δρόμου είναι A.



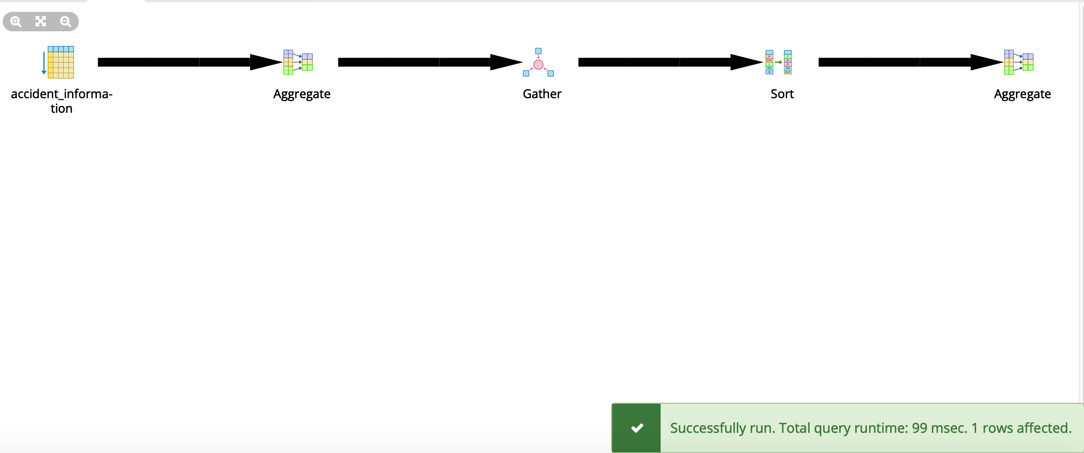
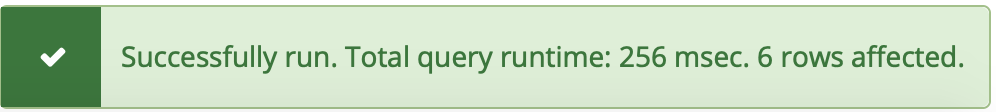
## Ερώτημα B

Στο δεύτερο ερώτημα θα ξανά εκτελέσουμε τα queries του προηγούμενου ερωτήματος αλλά αυτή τη φορά θα χρησιμοποιήσουμε ως buffer περισσότερη μνήμη από τη μνήμη RAM του υπολογιστή μας. Πιο συγκεκριμένα από τα 128ΜΒ που είναι το default της postgres θα το αυξήσουμε στο 1024ΜΒ. Έτσι μετά τη χρήση της εντολής **ALTER SYSTEM SET shared\_buffers TO ‘1024’MB** και επανεκκίνηση της postgres θα έχουμε το εξής αποτέλεσμα.

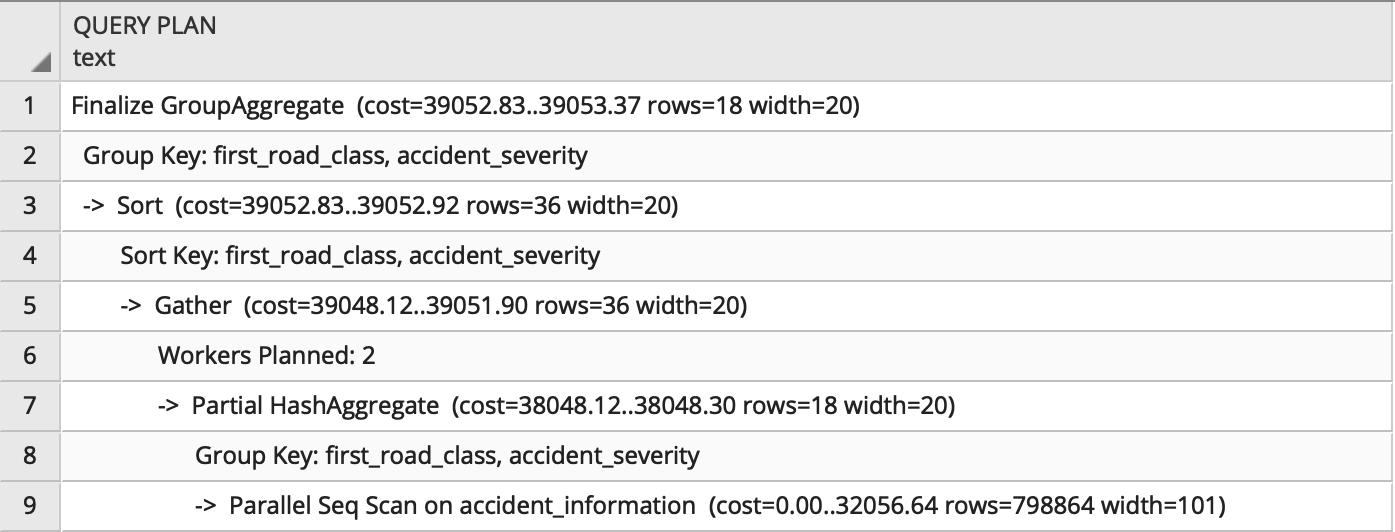
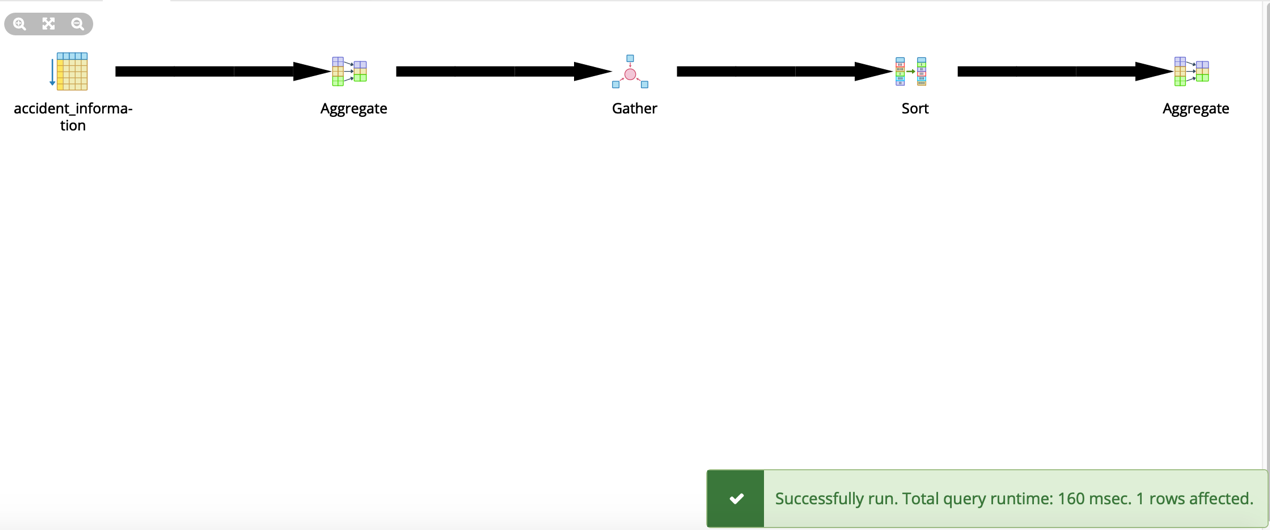
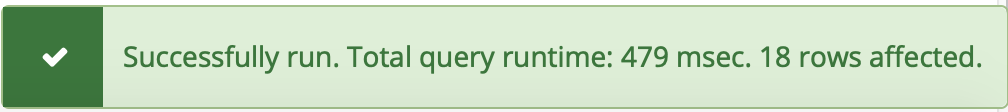


Έτσι έχουμε για κάθε query πρώτα το χρόνο εκτέλεσης, μετά την ανάλυση με EXPLAIN και τέλος το QUERY PLAN :

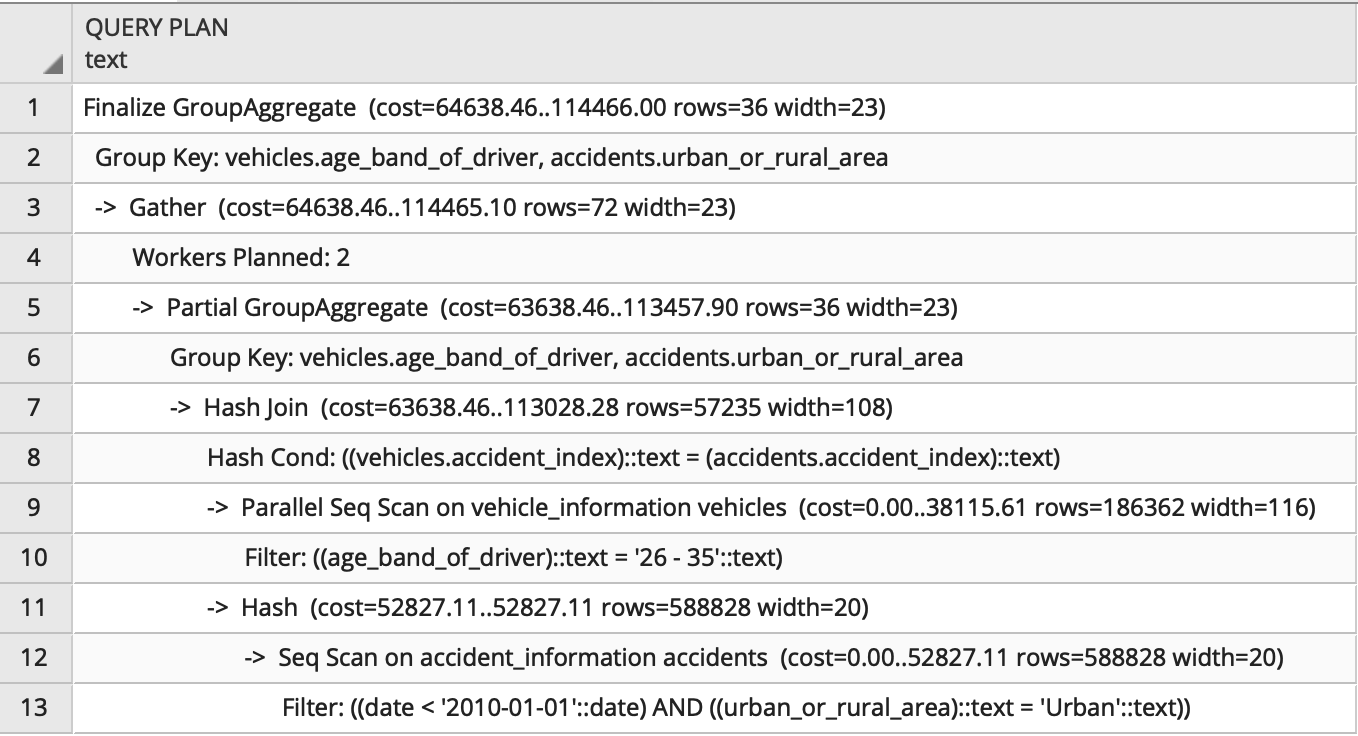
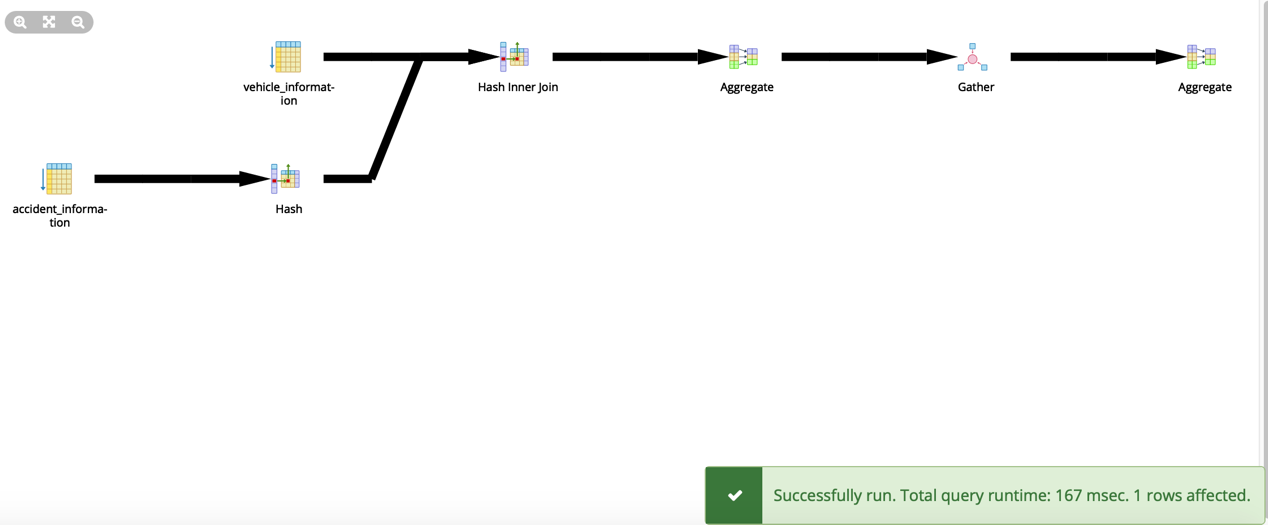
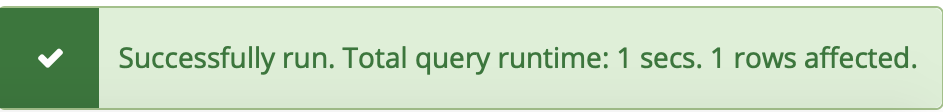
1. Πόσα ατυχήματα έγιναν ανά κατηγορία δρόμου (road class)



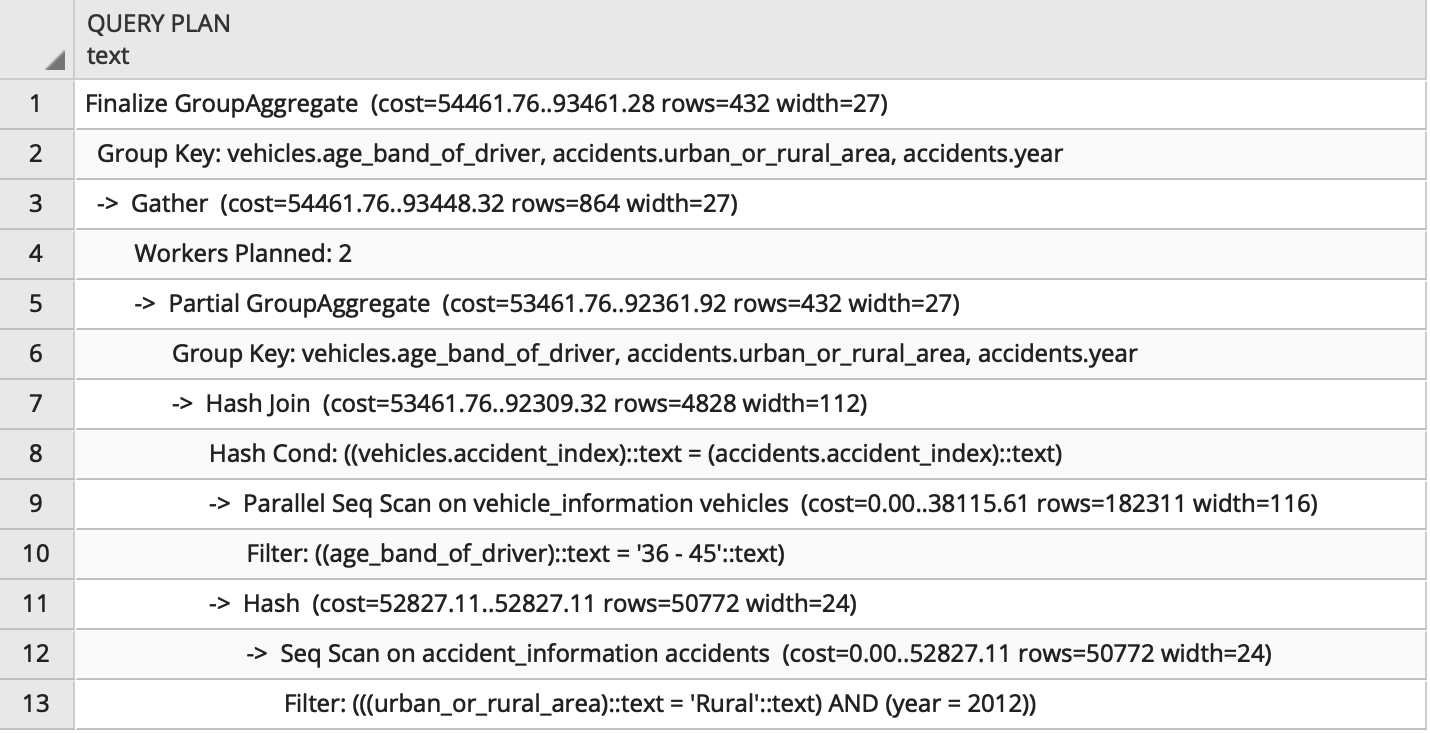
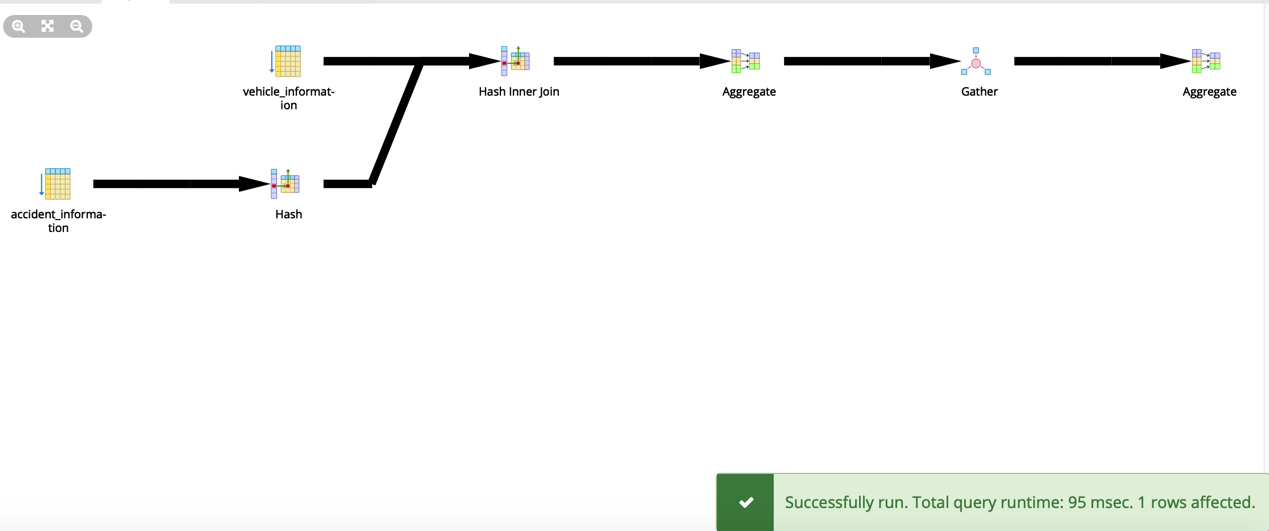
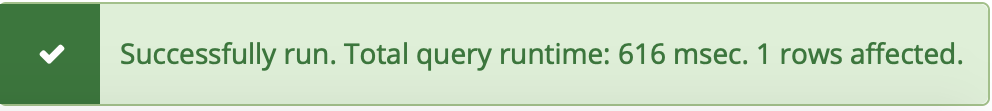
1. Πόσα είναι τα ατυχήματα ανά κατηγορία δρόμου και ανά κατηγορία ατυχήματος (Accident Severity).



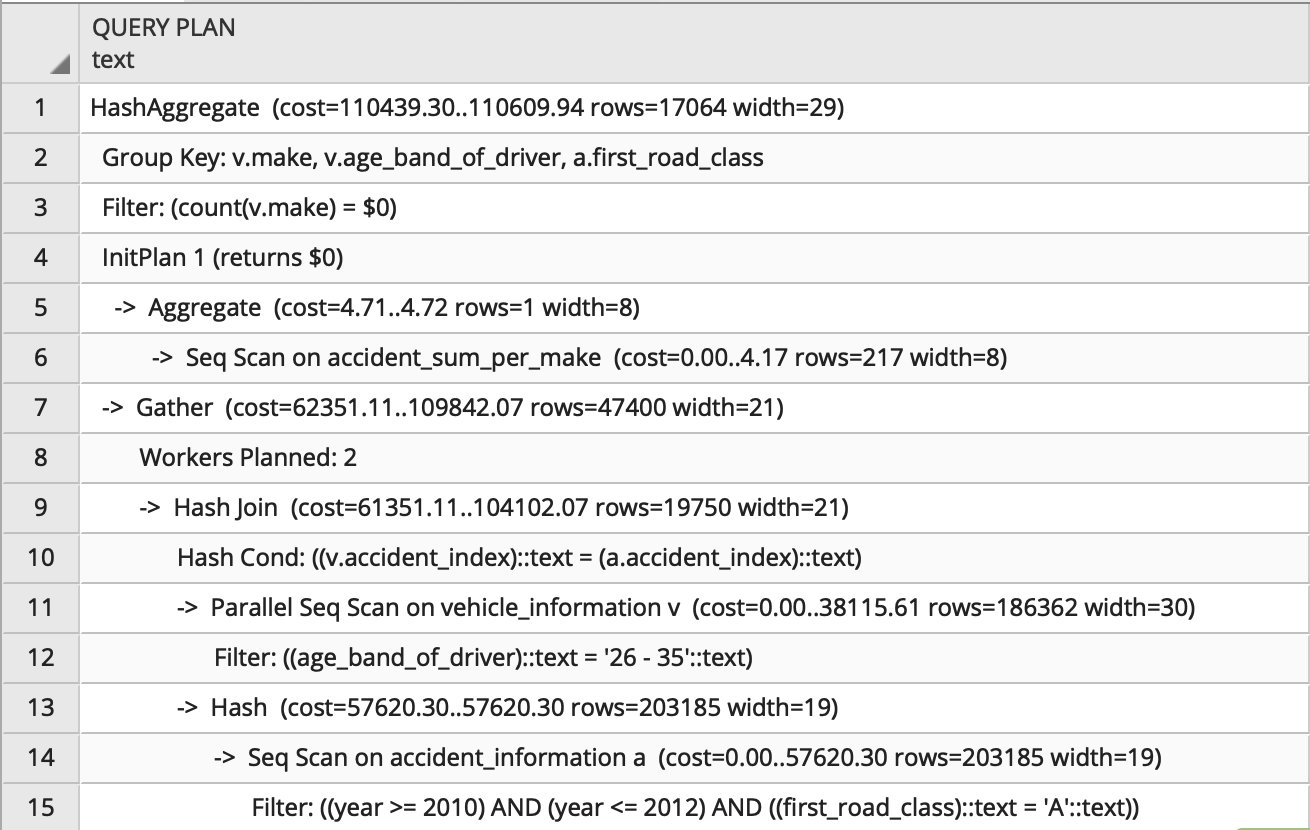
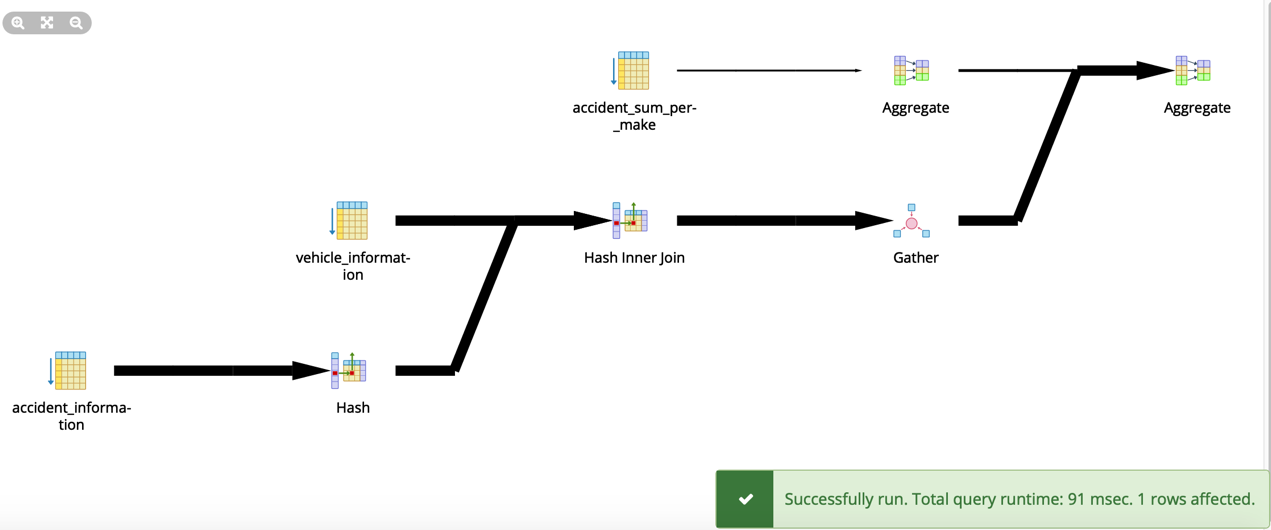
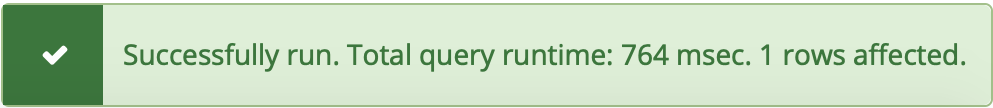
1. Πόσα ατυχήματα έγιναν σε αστική περιοχή πριν από την 1/1/2010 και η ηλικιακή κατηγορία του οδηγού είναι 26-35.



1. Πόσα ατυχήματα έγιναν σε αγροτική περιοχή το έτος 2012 και η ηλικιακή κατηγορία του οδηγού είναι 36-45.



1. Ποιος είναι ο κατασκευαστής του οχήματος που έχει τα περισσότερα ατυχήματα μεταξύ των ετών 2010 και 2012, η ηλικιακή κατηγορία των οδηγών είναι 26- 35 και η κατηγορία δρόμου είναι A.

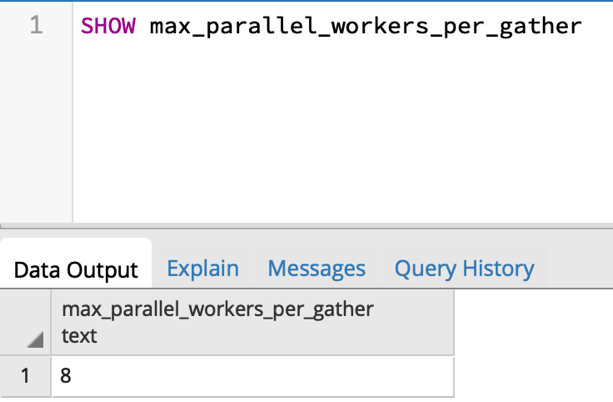


**Συμπέρασμα**

Όπως βλέπουμε υπάρχουν ελάχιστες βελτιώσεις στο χρόνο εκτέλεσης κάθε ερωτήματος και στα δύο τελευταία πολλές φορές ίσως και χειρότερο χρόνο εκτέλεσης. Άρα συμπεραίνουμε ότι ακόμα και αν τα shared\_buffers μας έχουν 1GB ελεύθερο χώρο για να φορτώσουμε ακόμα και ολόκληρα τα dataset στη μνήμη RAM δεν υπάρχει κάποια αξιοσημείωτη βελτίωση.

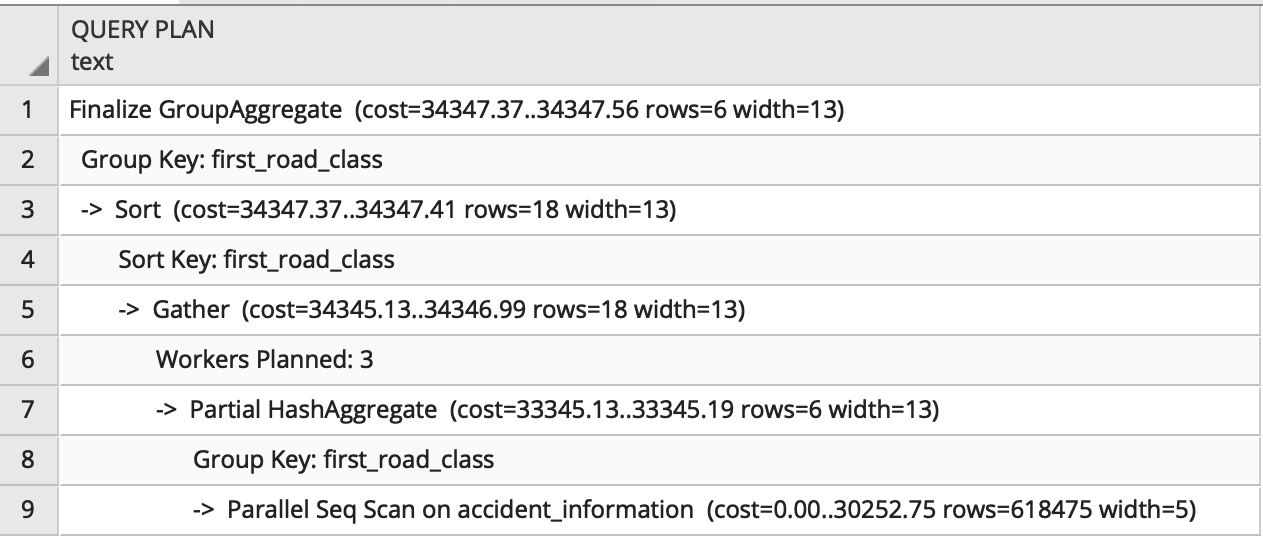
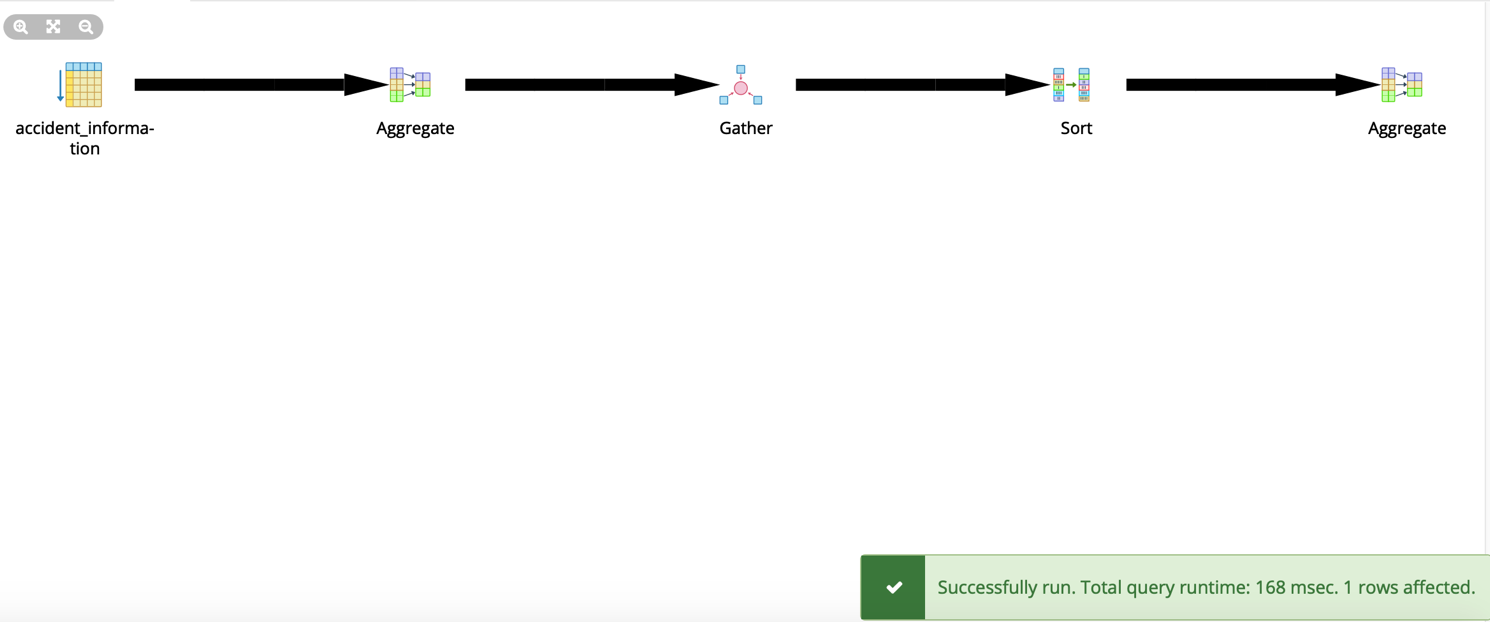
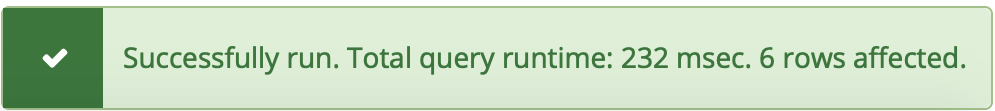
## Ερώτημα Γ

Στο τρίτο ερώτημα θα ξανά εκτελέσουμε τα queries του πρώτου ερωτήματος αλλά αυτή τη φορά θα χρησιμοποιήσουμε όλη την επεξεργαστική ισχύ του υπολογιστή μας. Πιο συγκεκριμένα από τους 2 parallel\_workers\_per\_gather που είναι το default της postgres θα το αυξήσουμε στους 8. Έτσι μετά τη χρήση της εντολής **ALTER SYSTEM SET max\_parallel\_workers\_per\_gather TO 8** και επανεκκίνηση της postgres θα έχουμε το εξής αποτέλεσμα.

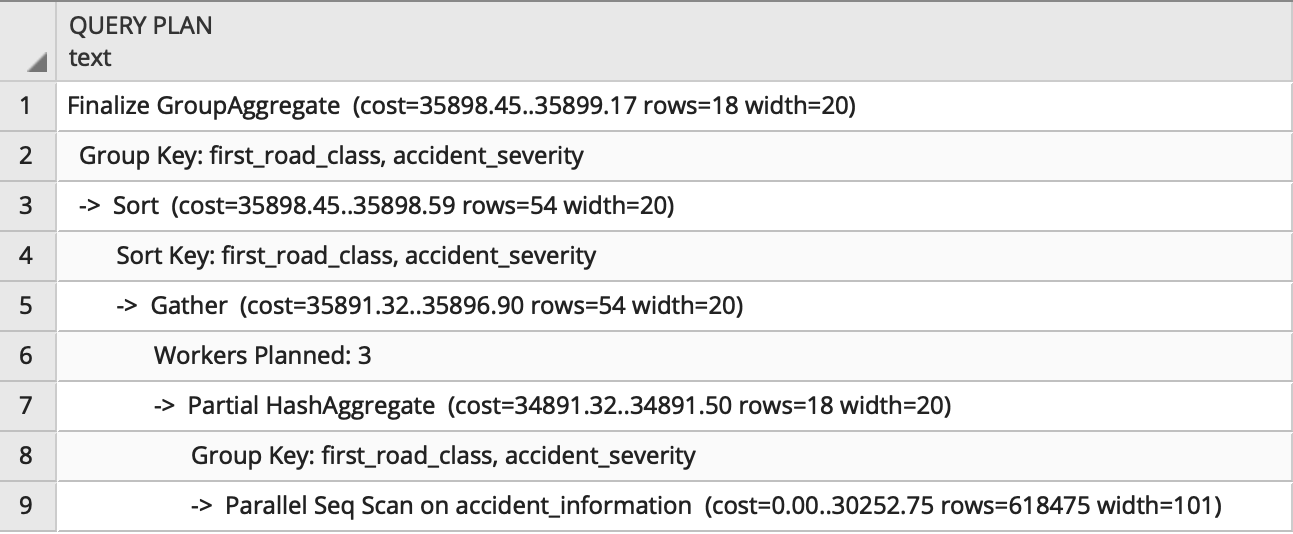
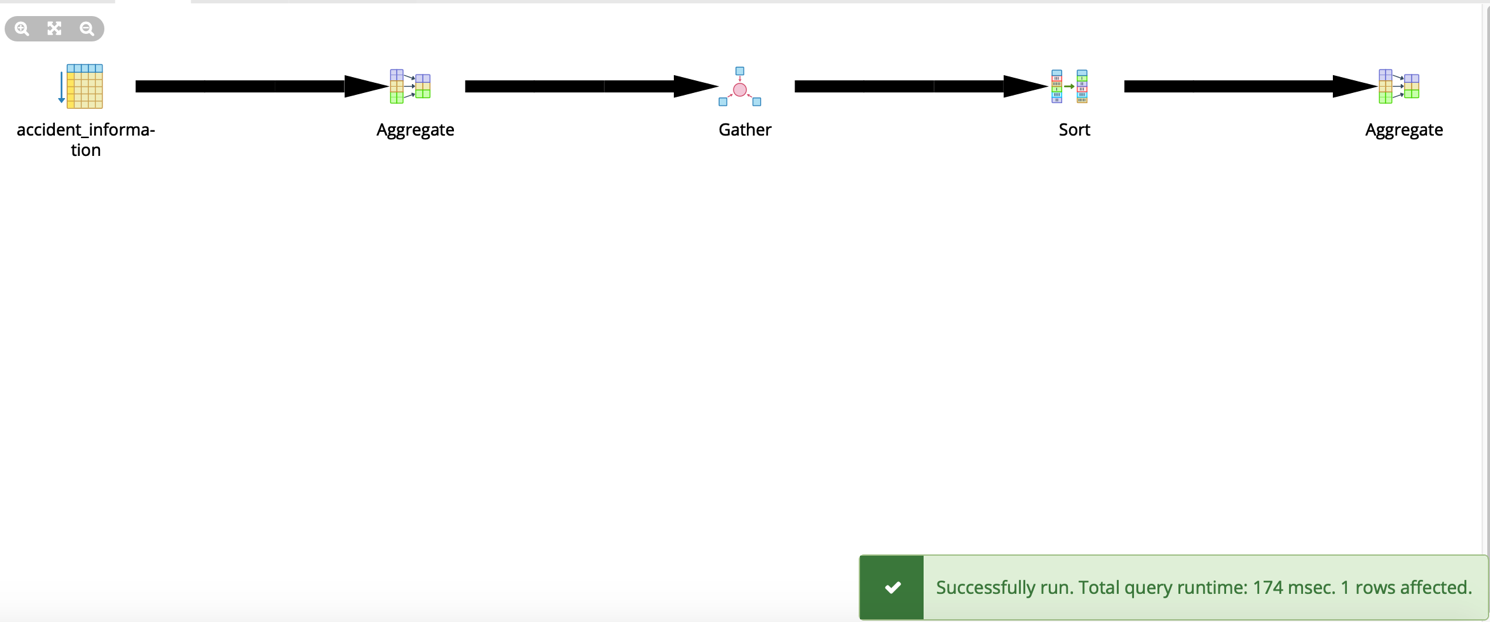
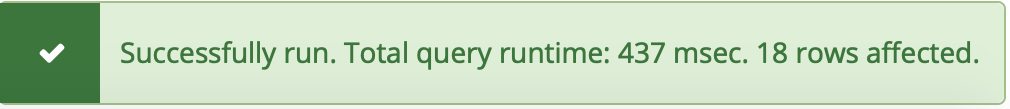


Έτσι έχουμε για κάθε query πρώτα το χρόνο εκτέλεσης, μετά την ανάλυση με EXPLAIN και τέλος το QUERY PLAN :

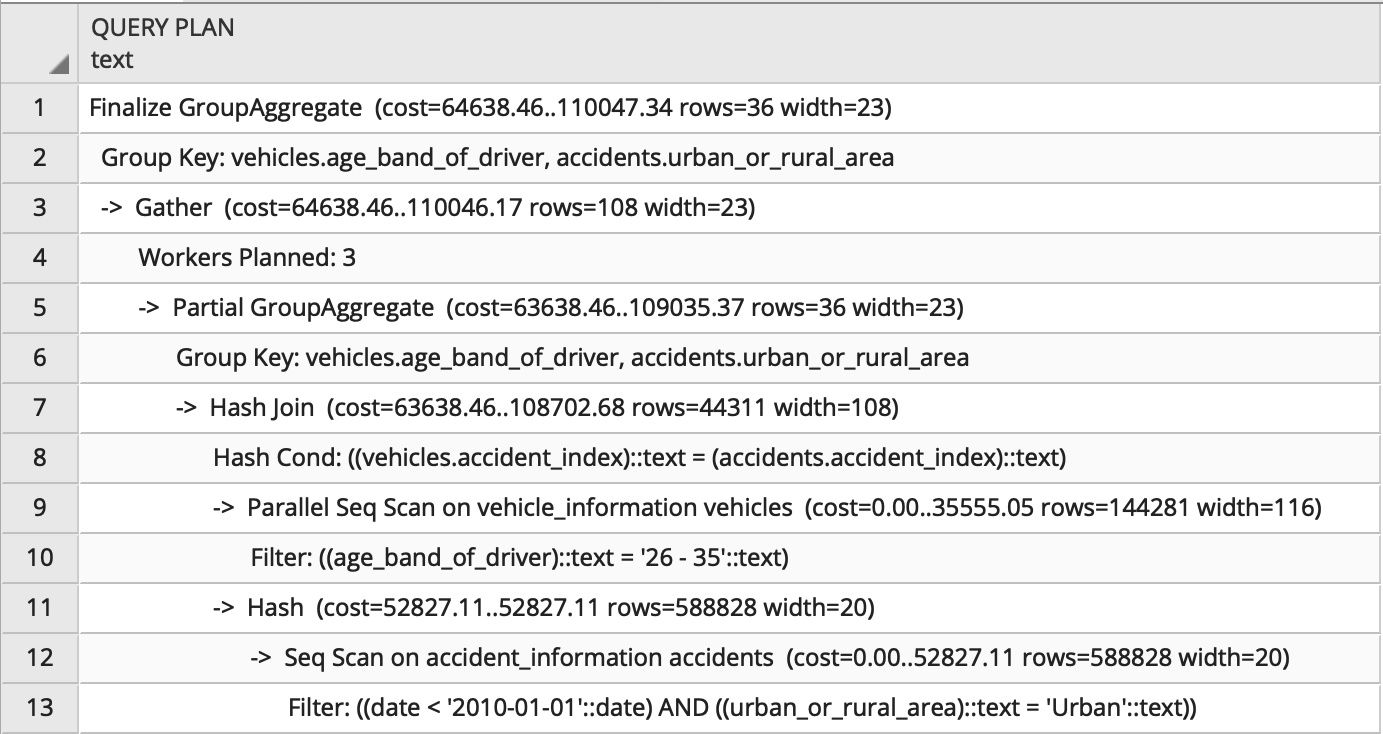
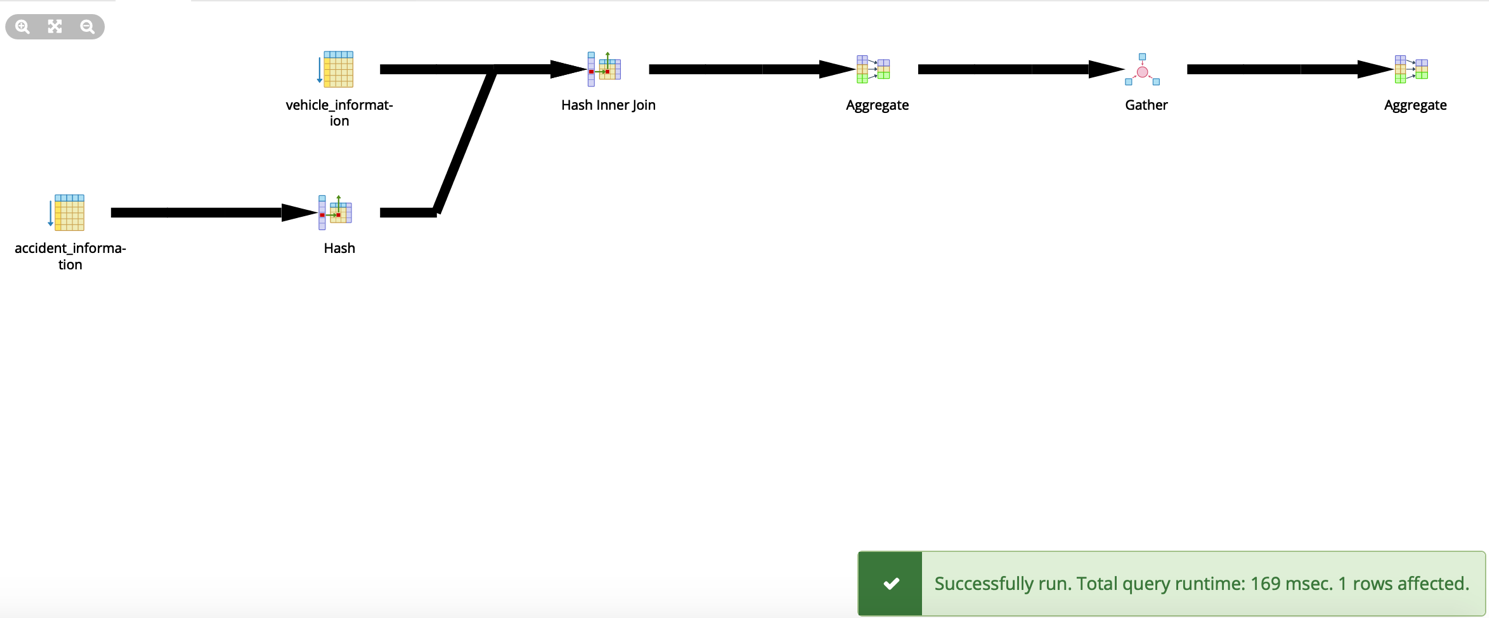
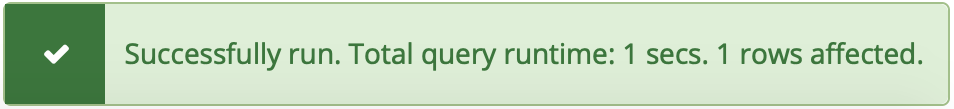
1. Πόσα ατυχήματα έγιναν ανά κατηγορία δρόμου (road class)



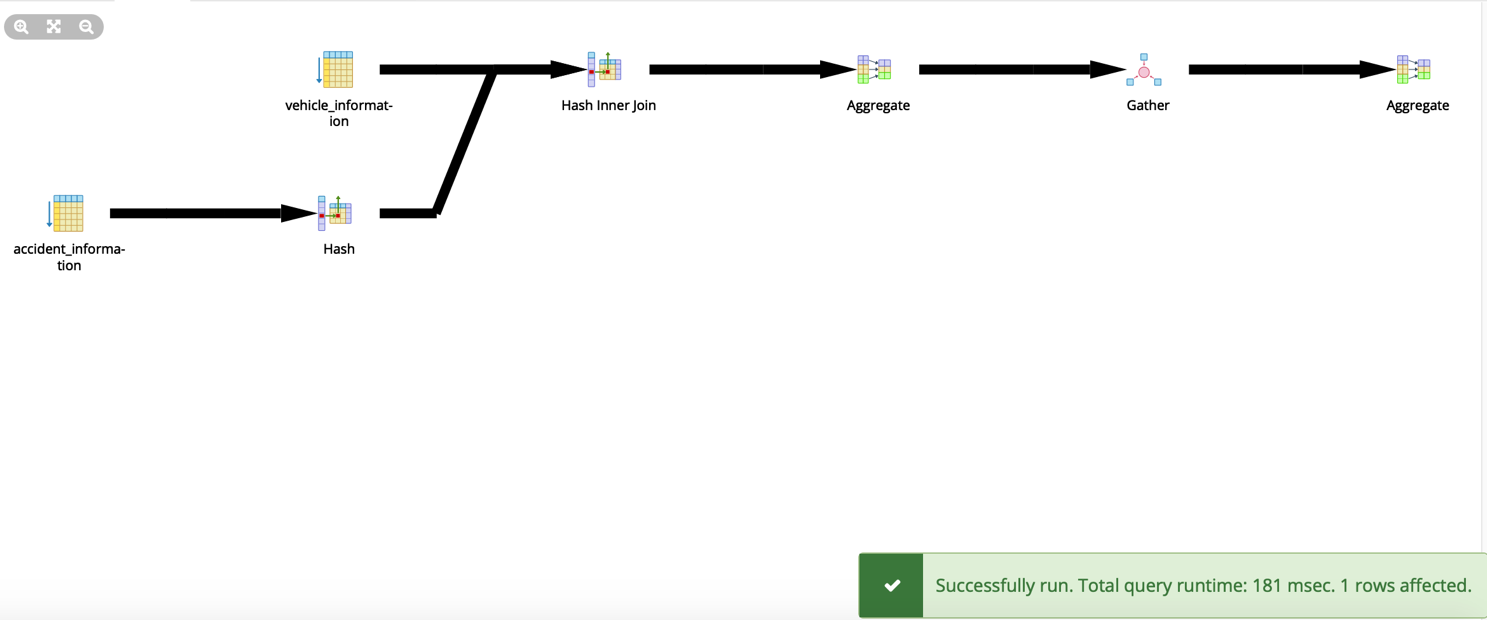
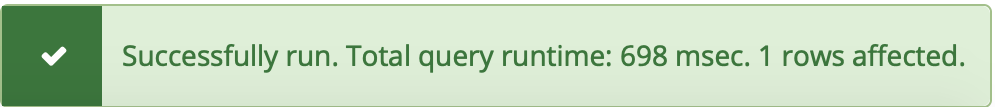
1. Πόσα είναι τα ατυχήματα ανά κατηγορία δρόμου και ανά κατηγορία ατυχήματος (Accident Severity).



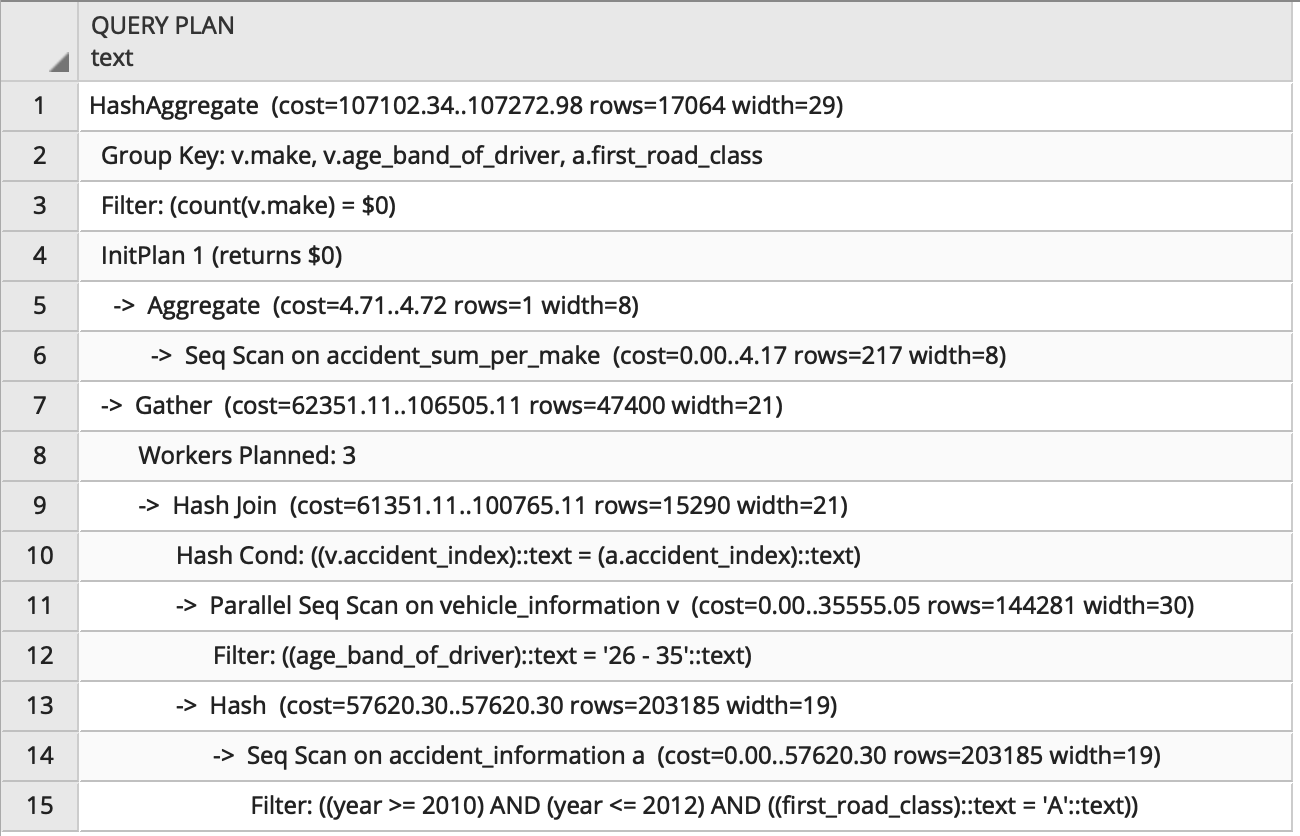
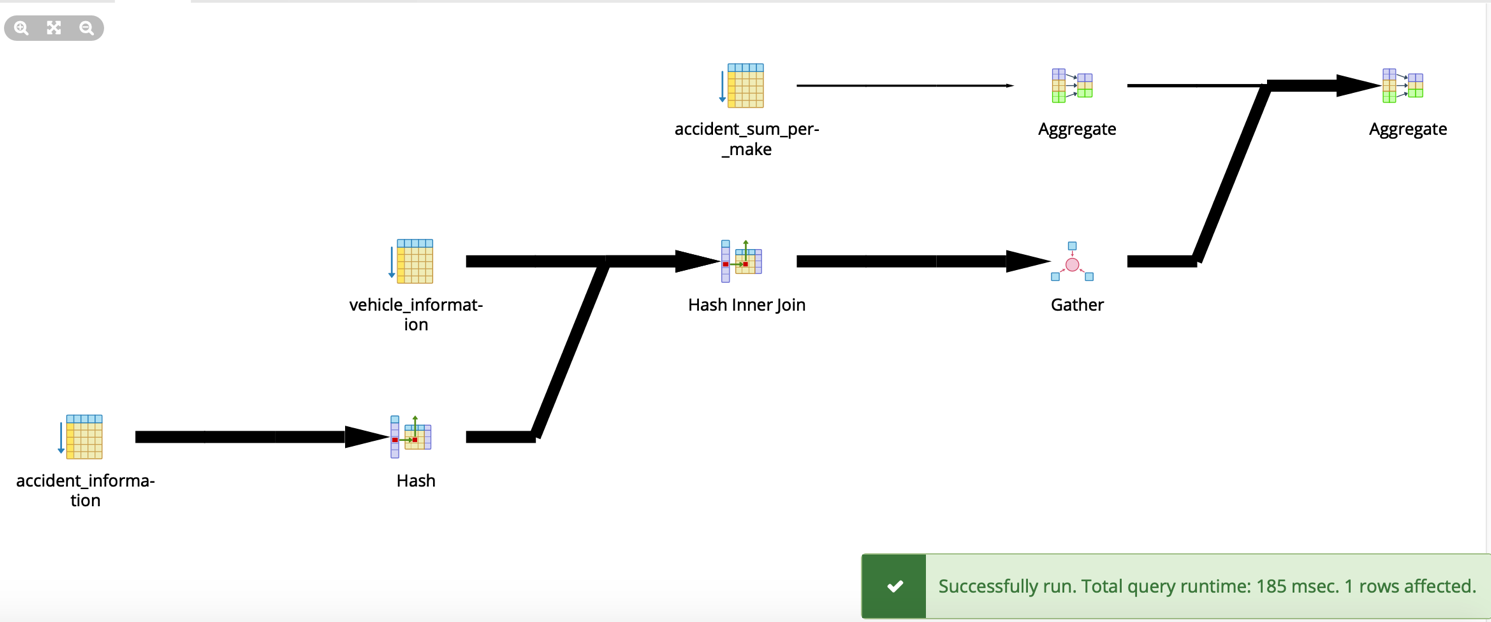
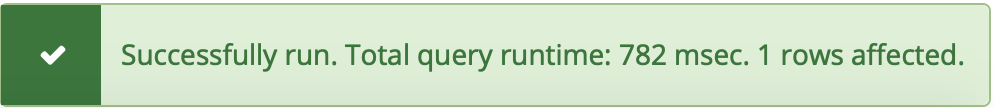
1. Πόσα ατυχήματα έγιναν σε αστική περιοχή πριν από την 1/1/2010 και η ηλικιακή κατηγορία του οδηγού είναι 26-35.



1. Πόσα ατυχήματα έγιναν σε αγροτική περιοχή το έτος 2012 και η ηλικιακή κατηγορία του οδηγού είναι 36-45.



1. Ποιος είναι ο κατασκευαστής του οχήματος που έχει τα περισσότερα ατυχήματα μεταξύ των ετών 2010 και 2012, η ηλικιακή κατηγορία των οδηγών είναι 26- 35 και η κατηγορία δρόμου είναι A.



**Συμπέρασμα**

Όπως βλέπουμε υπάρχουν σαφώς πιο εμφανίσιμες βελτιώσεις στο χρόνο εκτέλεσης κάθε ερωτήματος (σε σχέση με το ερώτημα 2) αν και στα δύο τελευταία queries ακόμα δεν παρατηρείται αξιοσημείωτη διαφορά όπως στα άλλα queries. Από την άλλη όμως υπάρχει αξιοσημείωτη διαφορά στο κόστος κάθε query (που το βλέπουμε στο query plan) καθώς τώρα από 2 workers planned ο υπολογιστής μας βάζει 3 (με μέγιστο το 8) προσφέροντας έτσι περισσότερη επεξεργαστική ισχύ και συνεπώς χαμηλότερο κόστος. Άρα συμπεραίνουμε ότι με όλη την επεξεργαστική ισχύ του υπολογιστή μας μπορούν να υπάρξουν βελτιώσεις στο χρόνο εκτέλεσης αλλά κυρίως θα υπάρξουν μεγαλύτερες βελτιώσεις στο κόστος του query.

## Ερώτημα Δ

Σε αυτή την ενότητα δημιουργούμε τα κατάλληλα ευρετήρια πάνω στους δύο πίνακες της βάσης δεδομένων, ώστε να επιταχύνουμε την απόκριση του συστήματος. Ας μελετήσουμε ένα προς ένα τα SQL queries του ερωτήματος Α συγκρίνοντας τα αποτελέσματα με βάση το χρόνο.

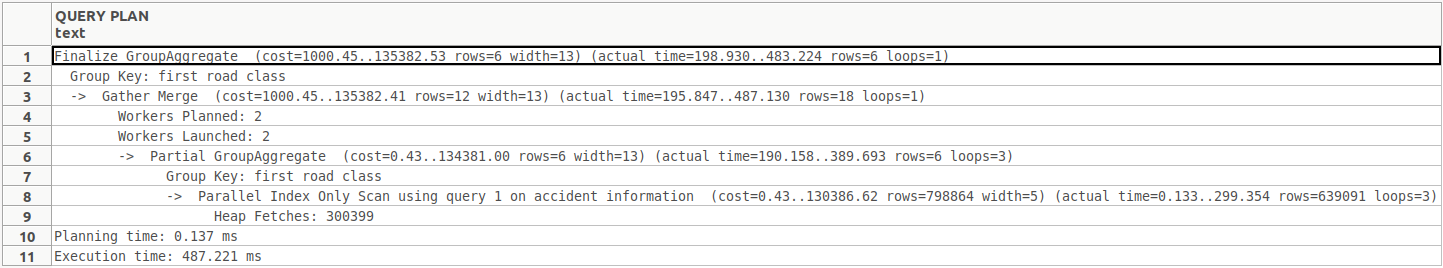
**Ερώτημα 1ο**: Βρείτε πόσα ατυχήματα έγιναν ανά κατηγορία δρόμου (road class).

Ανατρέχοντας στο αρχείο d\_1\_1.sql δημιουργούμε το index query\_1 στον πίνακα accident\_information και εκτελούμε ξανά το query του αρχείου a\_1\_1.sql. Το index που χρησιμοποιούμε βασίζεται στη δομή B-Tree για να επιταχύνουμε την ομαδοποίηση του GROUP BY πάνω στο first\_road\_class.

Γραφικό σχήμα επεξήγησης:



Απλό σχήμα επεξήγησης:



Παρατηρούμε ότι από 781ms μειώθηκε ο χρόνος εκτέλεσης σε **487ms**.

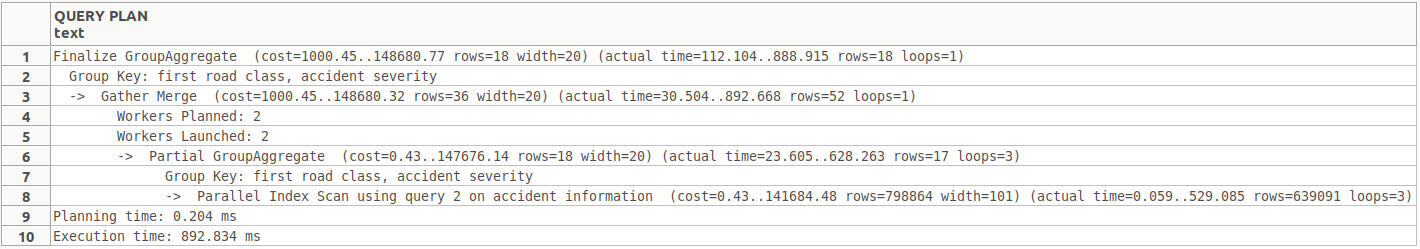
**Ερώτημα 2ο**: Πόσα είναι τα ατυχήματα ανά κατηγορία δρόμου και ανά κατηγορία ατυχήματος (Accident Severity).

Ανατρέχοντας στο αρχείο d\_1\_2.sql δημιουργούμε το index query\_2 στον πίνακα accident\_information και εκτελούμε ξανά το query του αρχείου a\_1\_2.sql. Το index που χρησιμοποιούμε βασίζεται στη δομή B-Tree για να επιταχύνουμε την ομαδοποίηση του GROUP BY πάνω στα first\_road\_class και accident\_severity.

Γραφικό σχήμα εκτέλεσης:



Απλό σχήμα εκτέλεσης:



Παρατηρούμε ότι από 256ms αυξήθηκε ο χρόνος εκτέλεσης σε **892ms**.

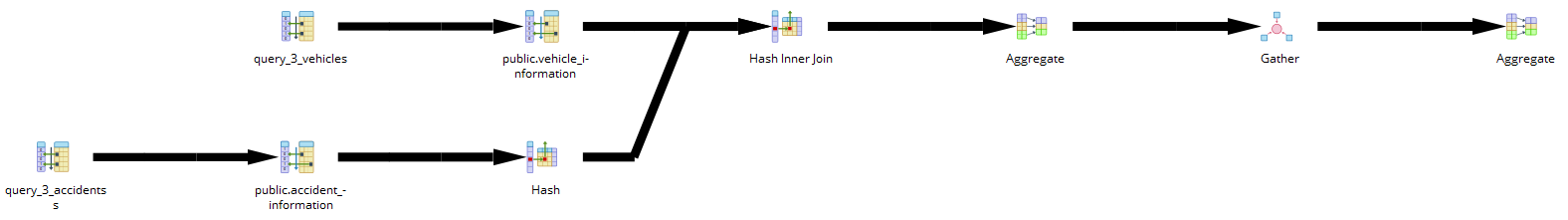
**Ερώτημα 3ο**: Βρείτε πόσα ατυχήματα έγιναν σε αστική περιοχή πριν από την 1/1/2010 και η ηλικιακή κατηγορία του οδηγού είναι 26-35.

Ανατρέχοντας στο αρχείο d\_1\_3.sql δημιουργούμε δυο indexes:

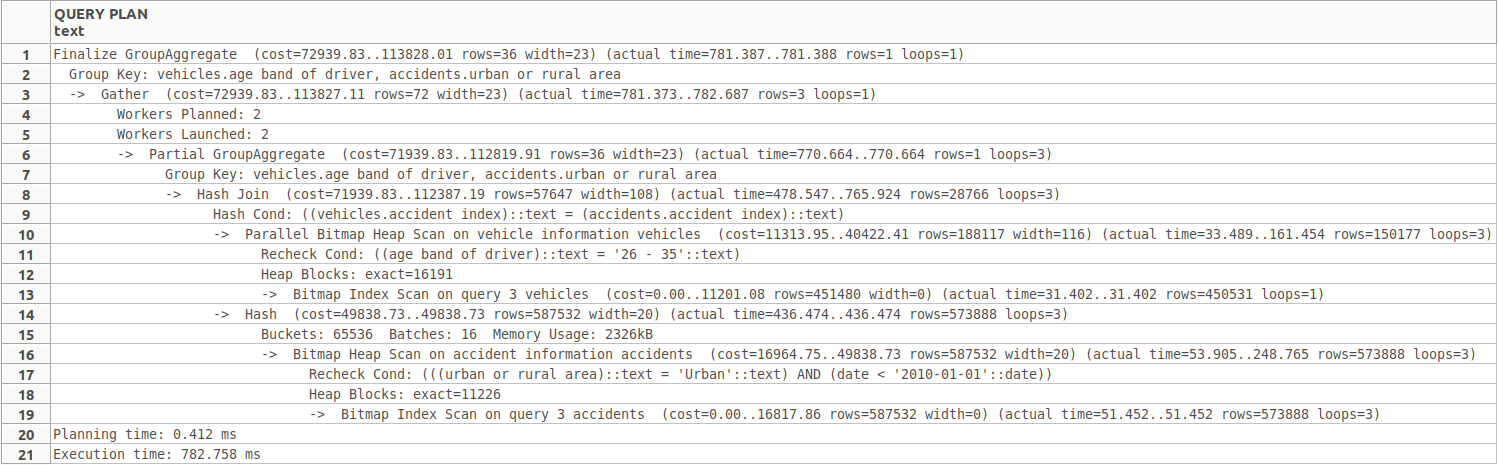
* Το query\_3\_accidents πάνω στον πίνακα accident\_information με τα πεδία accident\_index, date και urban\_or\_rural\_area. Πρόσθετοι περιορισμοί το urban\_or\_rural area πρέπει να έχει τη τιμή ‘Urban’ και το date πρέπει να είναι μικρότερο από ‘2010-1-1’.
* Το query\_3\_vehicles πάνω στον πίνακα vehicles\_information με τα πεδία accident\_index και age\_band\_of\_driver. Πρόσθετος περιορισμός το age\_band\_of\_driver πρέπει να έχει τιμή ‘26 – 35’.

Εκτελούμε ξανά το query του αρχείου a\_1\_3.sql. Και στα δυο indexes ακολουθούμε τη δομή B-Tree εφόσον έχουμε συγκρίσεις και ομαδοποίηση με INNER JOIN, WHERE και GROYP BY. Επιπρόσθετα, περιορίζουμε το μέγεθος των index στις τιμές που ενδιαφέρουν το query του ερωτήματος.

Γραφικό σχήμα εκτέλεσης:



Απλό σχήμα εκτέλεσης:



Παρατηρούμε ότι από 1s μειώθηκε ο χρόνος εκτέλεσης σε **782ms**.

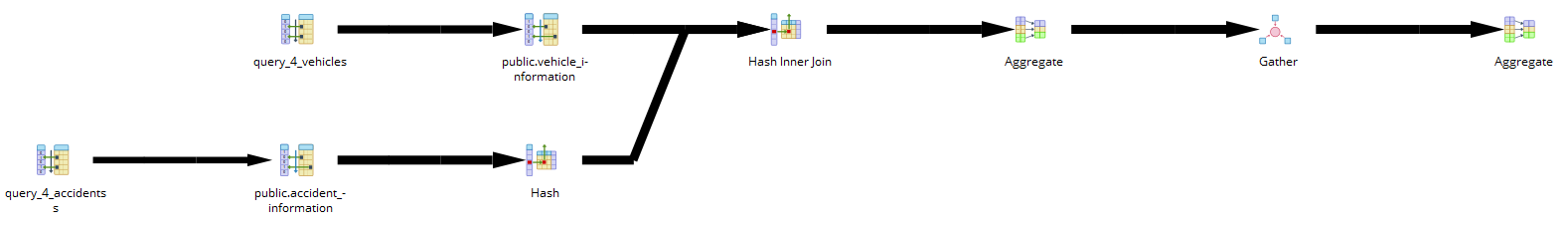
**Ερώτημα 4ο**: Βρείτε πόσα ατυχήματα έγιναν σε αγροτική περιοχή το έτος 2012 και η ηλικιακή κατηγορία του οδηγού είναι 36-45.

Ανατρέχοντας στο αρχείο d\_1\_4.sql δημιουργούμε δυο indexes:

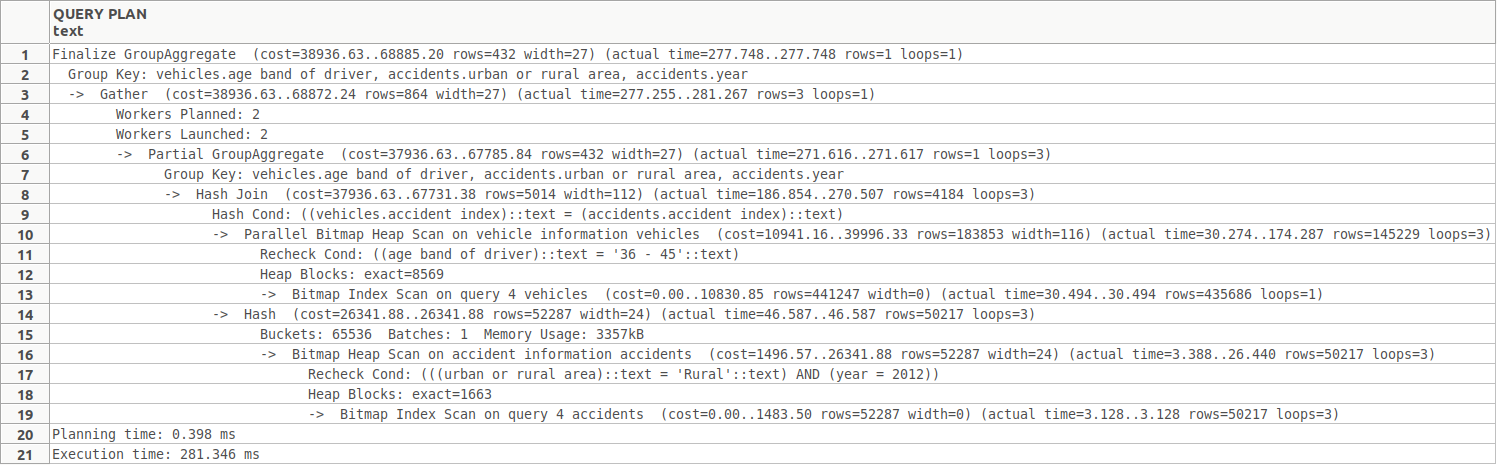
* Το query\_4\_accidents πάνω στον πίνακα accident\_information με τα πεδία accident\_index, year και urban\_or\_rural\_area. Πρόσθετοι περιορισμοί το urban\_or\_rural\_area πρέπει να έχει τη τιμή ‘Rural’ και το year πρέπει να είναι ’2012'.
* Το query\_4\_vehicles πάνω στον πίνακα vehicles\_information με τα πεδία accident\_index και age\_band\_of\_driver. Πρόσθετος περιορισμός το age\_band\_of\_driver πρέπει να έχει τιμή ‘36 – 45’.

Εκτελούμε ξανά το query του αρχείου a\_1\_4.sql. Και στα δυο indexes ακολουθούμε τη δομή B-Tree εφόσον έχουμε συγκρίσεις και ομαδοποίηση με INNER JOIN, WHERE και GROYP BY. Επιπρόσθετα, περιορίζουμε το μέγεθος των index στις τιμές που ενδιαφέρουν το query του ερωτήματος.

Γραφικό σχήμα εκτέλεσης:



Απλό σχήμα εκτέλεσης:



Παρατηρούμε ότι από 616ms μειώθηκε ο χρόνος εκτέλεσης σε **281ms**.

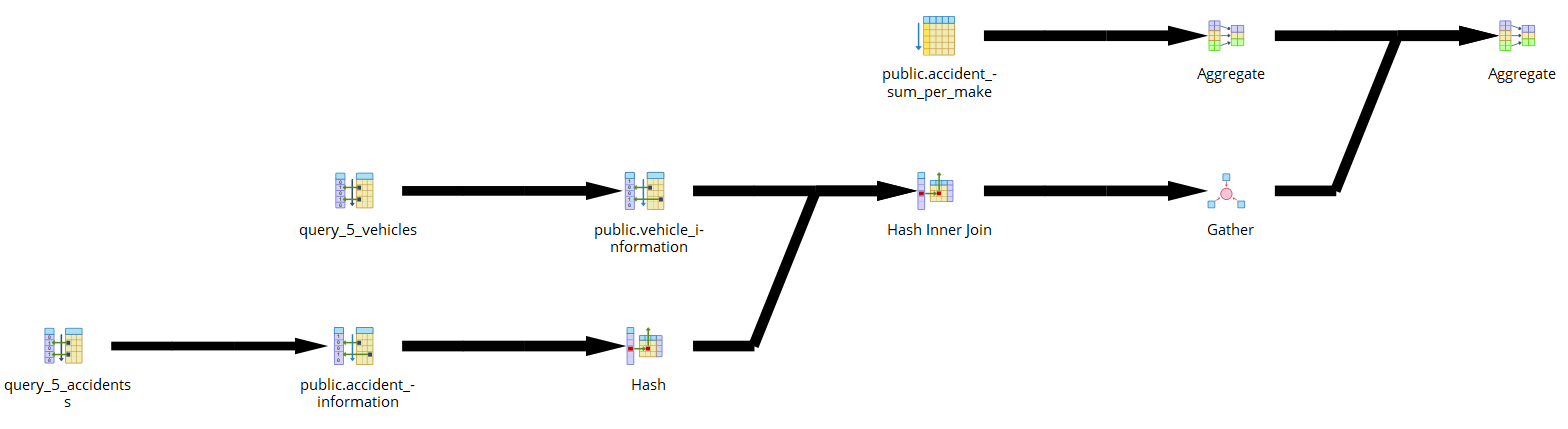
**Ερώτημα 5ο**: Βρείτε τον κατασκευαστή του οχήματος που έχει τα περισσότερα ατυχήματα μεταξύ των ετών 2010 και 2012, ηλικιακή κατηγορία των οδηγών είναι 26-35 και η κατηγορία δρόμου είναι Α.

Ανατρέχοντας στο αρχείο d\_1\_5.sql δημιουργούμε δυο indexes:

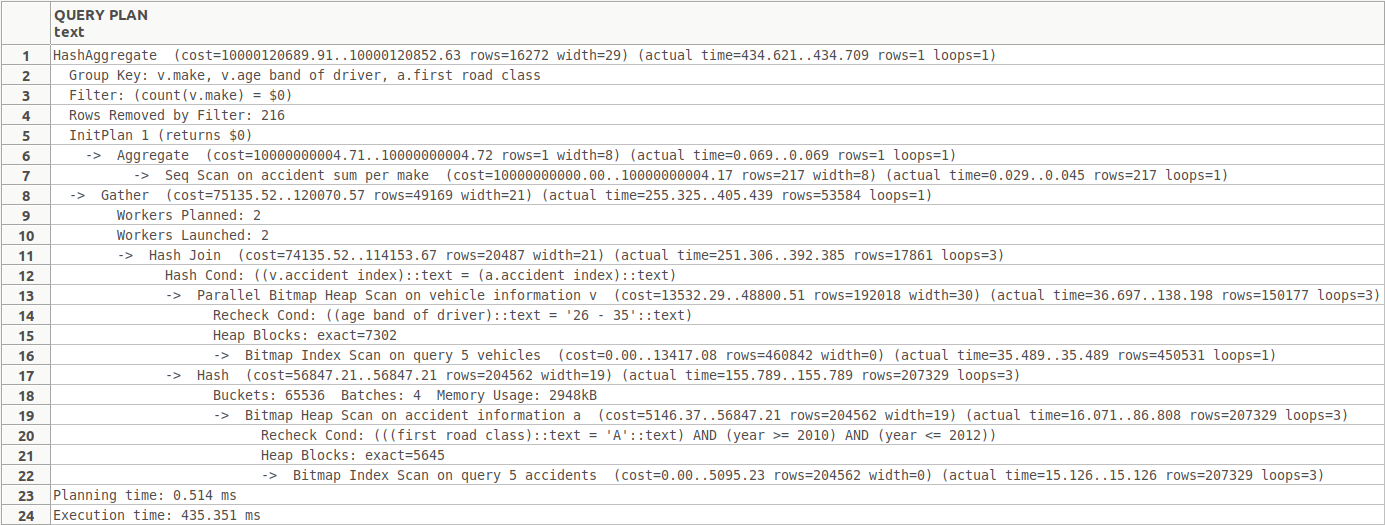
* Το query\_5\_accidents πάνω στον πίνακα accident\_information με τα πεδία accident\_index, first\_road\_class και year. Πρόσθετοι περιορισμοί το first\_road\_class πρέπει να έχει τη τιμή ‘A’ και το year να είναι ανάμεσα στο 2010 και 2012.
* Το query\_5\_vehicles πάνω στον πίνακα vehicles\_information με τα πεδία accident\_index και age\_band\_of\_driver και make. Πρόσθετος περιορισμός το age\_band\_of\_driver πρέπει να έχει τιμή ‘26 – 35’.

Εκτελούμε ξανά το query του αρχείου a\_1\_5.sql. Και στα δυο indexes ακολουθούμε τη δομή B-Tree εφόσον έχουμε συγκρίσεις και ομαδοποίηση με INNER JOIN, WHERE και GROYP BY. Επιπρόσθετα, περιορίζουμε το μέγεθος των index στις τιμές που ενδιαφέρουν το query του ερωτήματος.

Γραφικό σχήμα εκτέλεσης:



Απλό σχήμα εκτέλεσης:



Παρατηρούμε ότι από 781ms μειώθηκε ο χρόνος εκτέλεσης σε **435ms**.

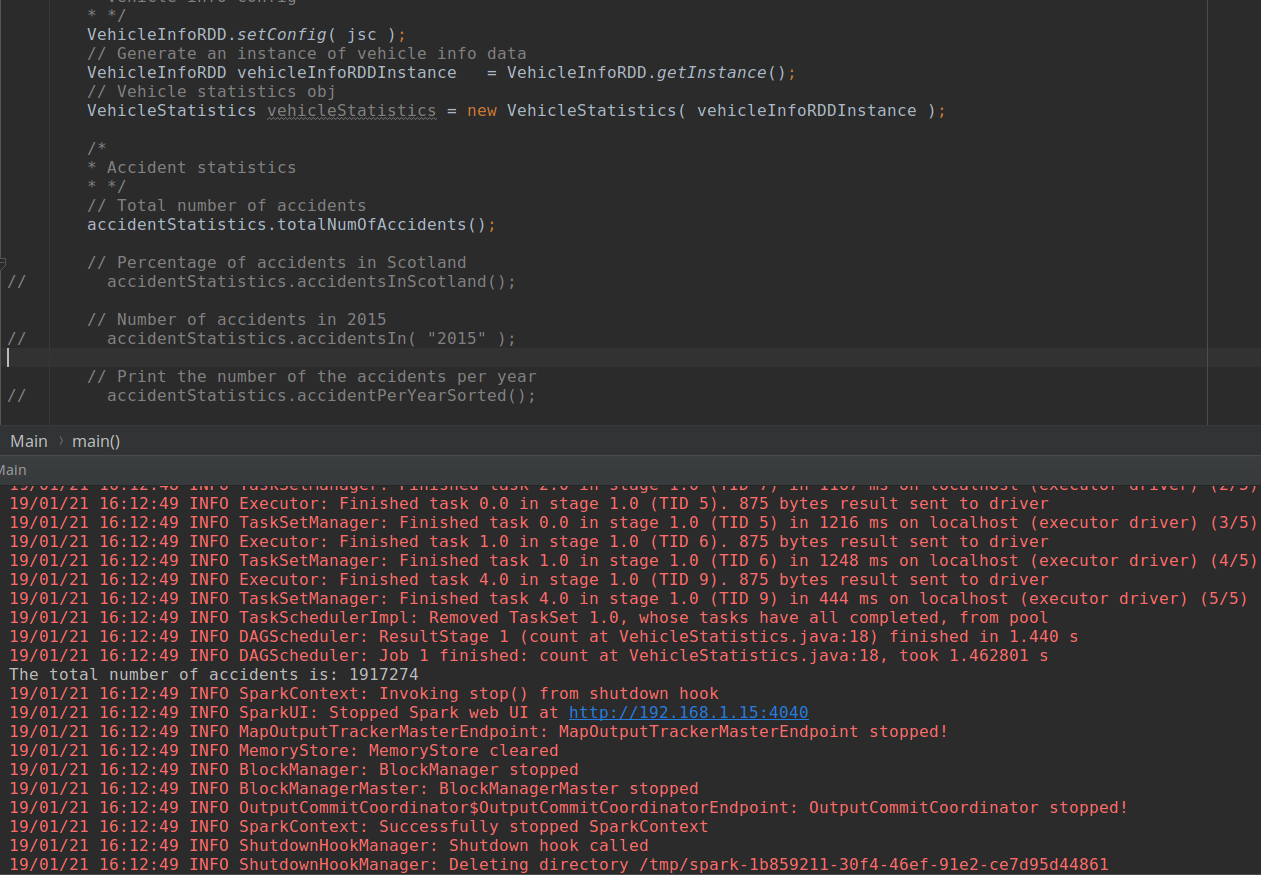
## Ερώτημα Ε

# **Μέρος 2**

## Ερώτημα Α

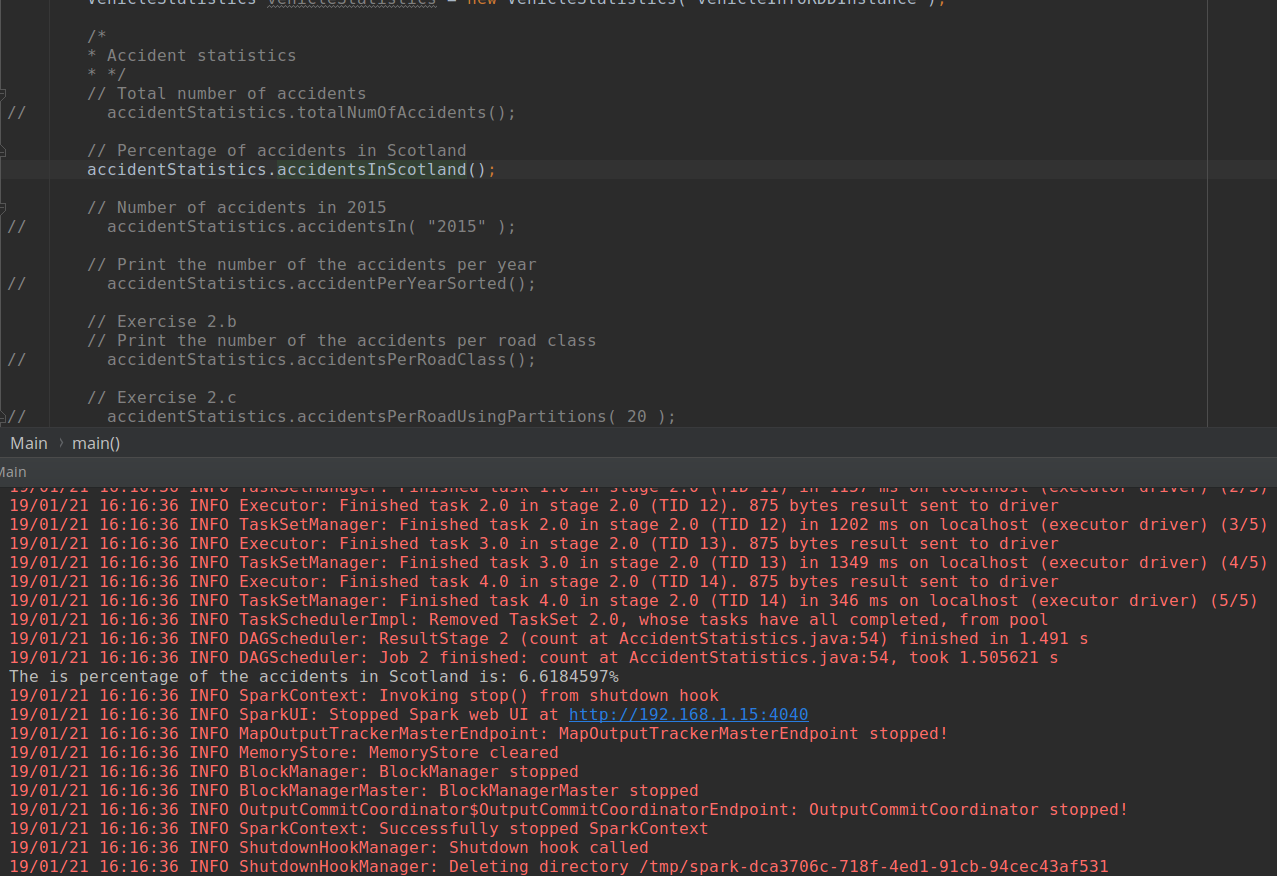
Ακολουθώντας τις οδηγίες ,παρουσιάζονται κάποια στατιστικά δεδομένα.

1. Στην εικόνα 2.1 παρουσιάζεται ο συνολικός αριθμός των ατυχημάτων που συνέβησαν. Όπως φαίνεται ο συνολικός αριθμός των ατυχημάτων είναι 1917274.



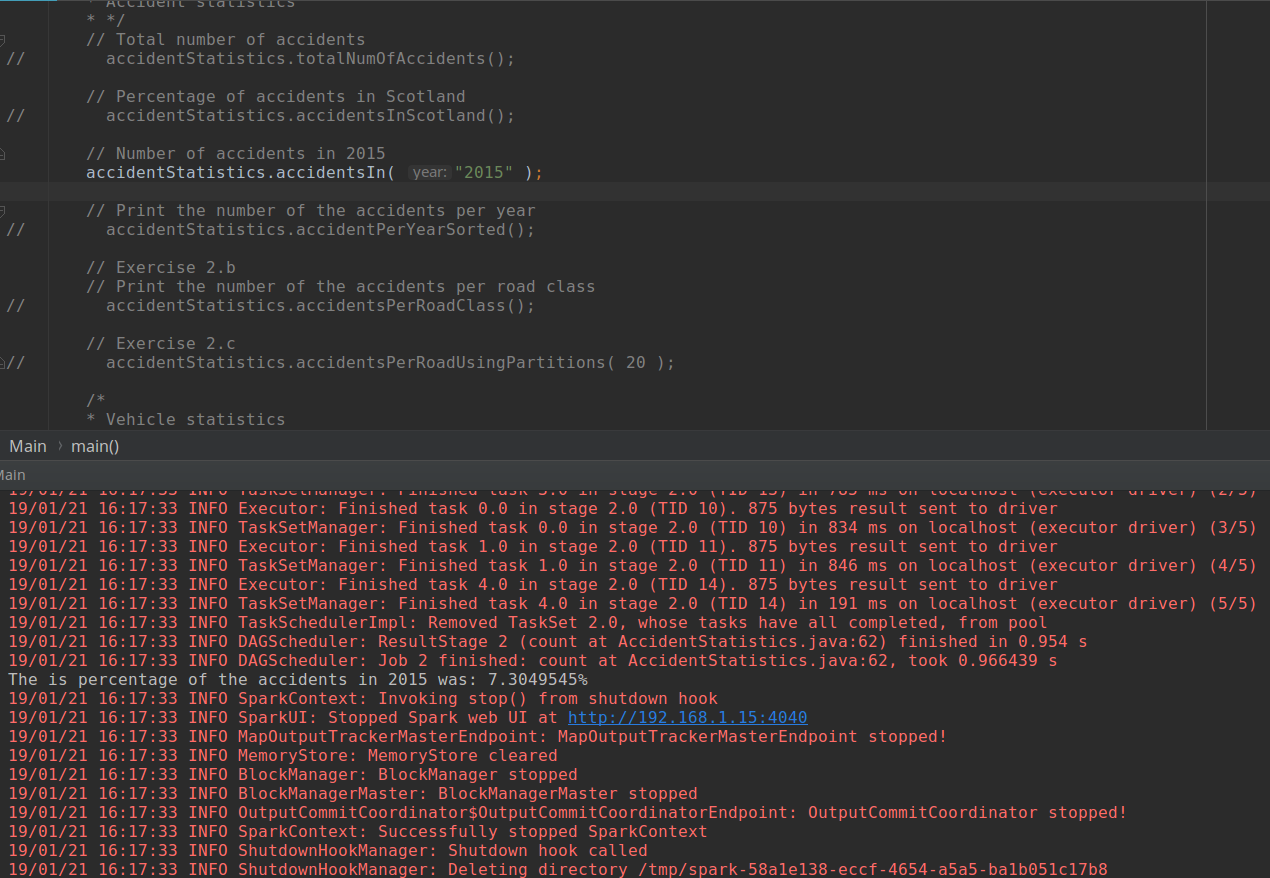
Εικόνα 2.2.1

1. Στην εικόνα 2.2 παρουσιάζεται το ποσοστό επι τις εκατό των ατυχημάτων που συνέβησαν στη Σκωτία, το οποίο είναι 6,6184597%.



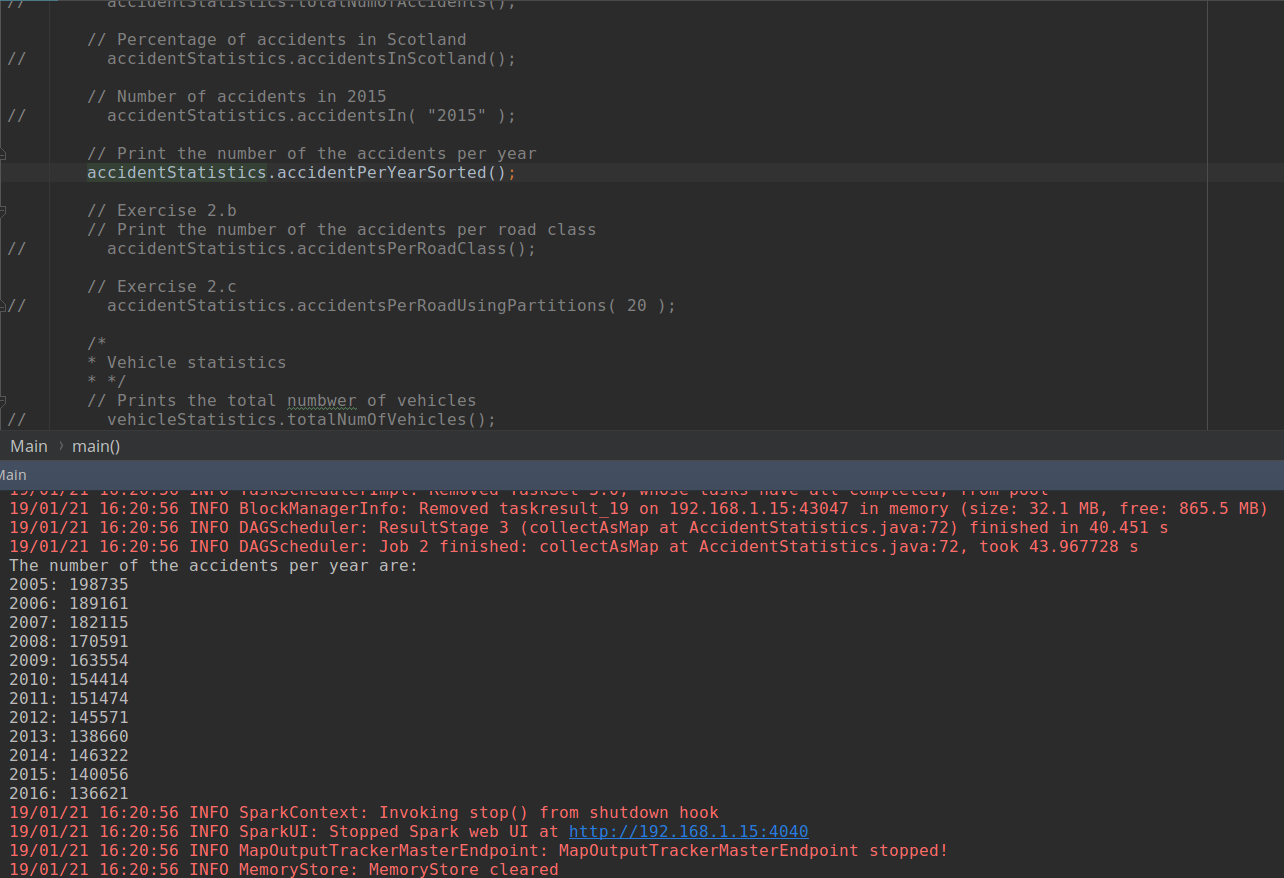
Εικόνα 2.2.2

1. Στην εικόνα 2.3 παρουσιάζεται το ποσοστό επι της εκατό των ατυχημάτων που έγιναν το 2015. Όπως φαίνεται, στη μέθοδο accidentsIn μπορούμε να δώσουμε ως όρισμα το έτος που θέλουμε και να εμφανιστούν τα αντίστοιχα αποτελέσματα. Για χάρη το παραδείγματος εμείς παρουσιάζουμε τα στοιχεία για το 2015 και αυτά είναι 7,3049545%.



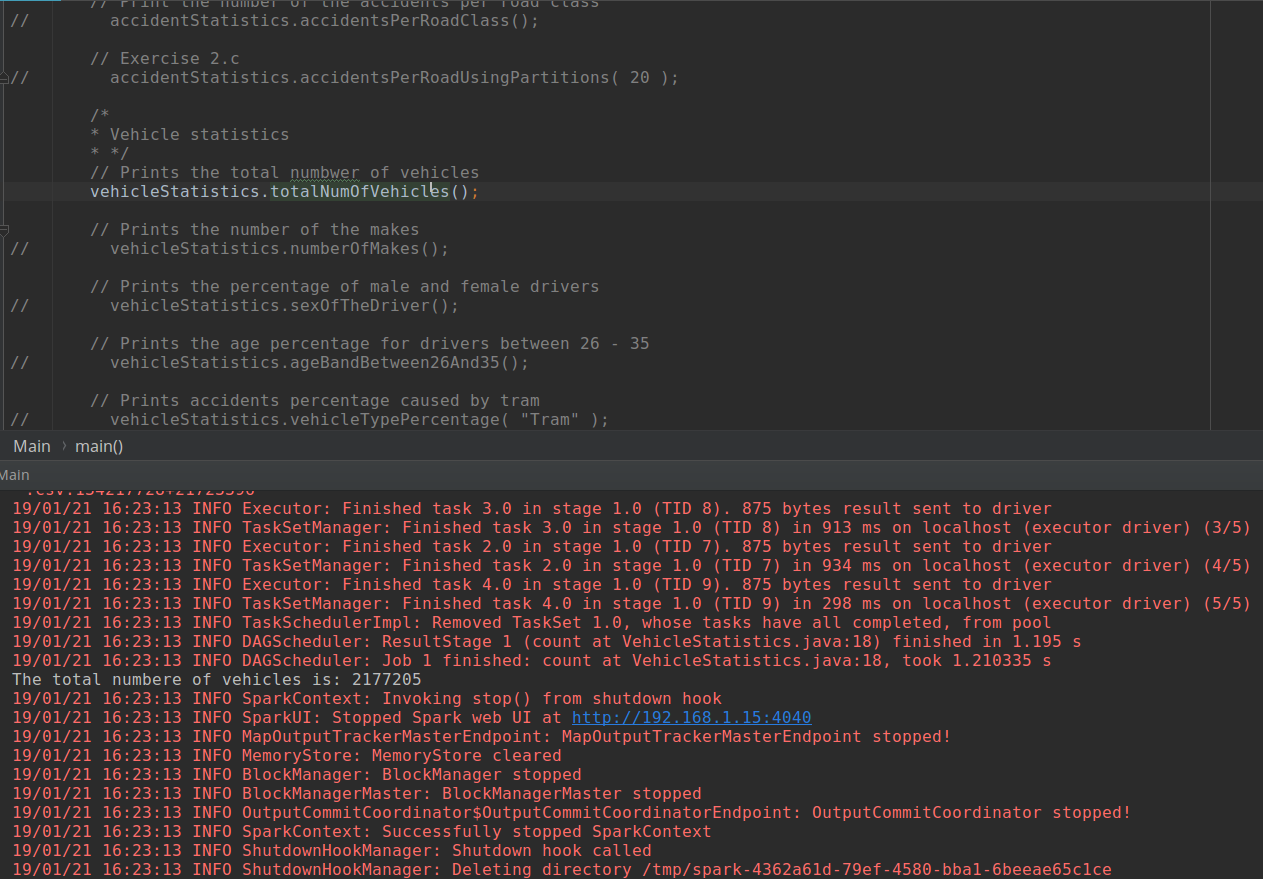
Εικόνα 2.2.3

1. Στην εικόνα 2.4 παρουσιάζεται το σύνολο των ατυχημάτων που συνέβησαν ανά χρόνο.



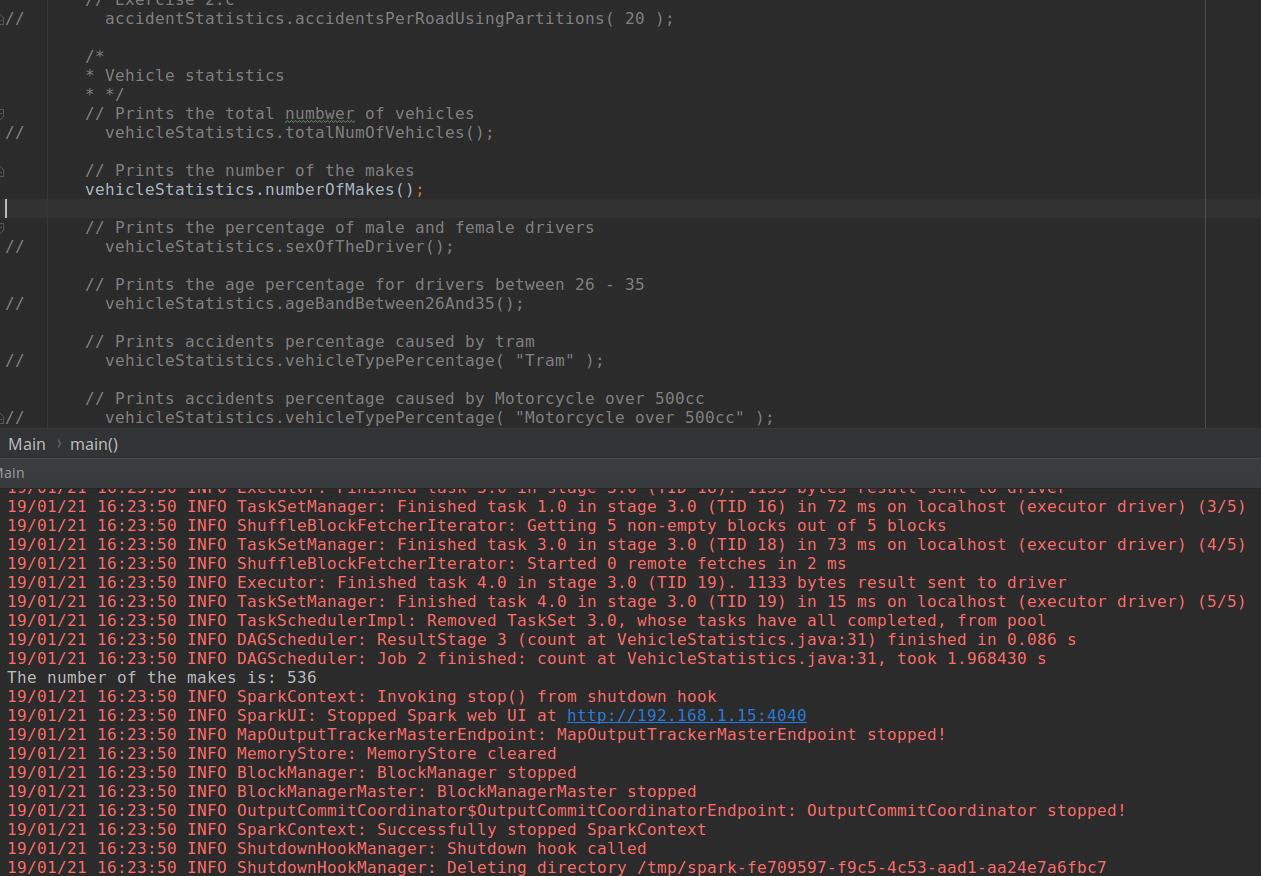
Εικόνα 2.2.4

1. Στην εικόνα 2.5 παρουσιάζεται ο συνολικός αριθμός των αυτοκινήτων, ο οποίος είναι 2177205.



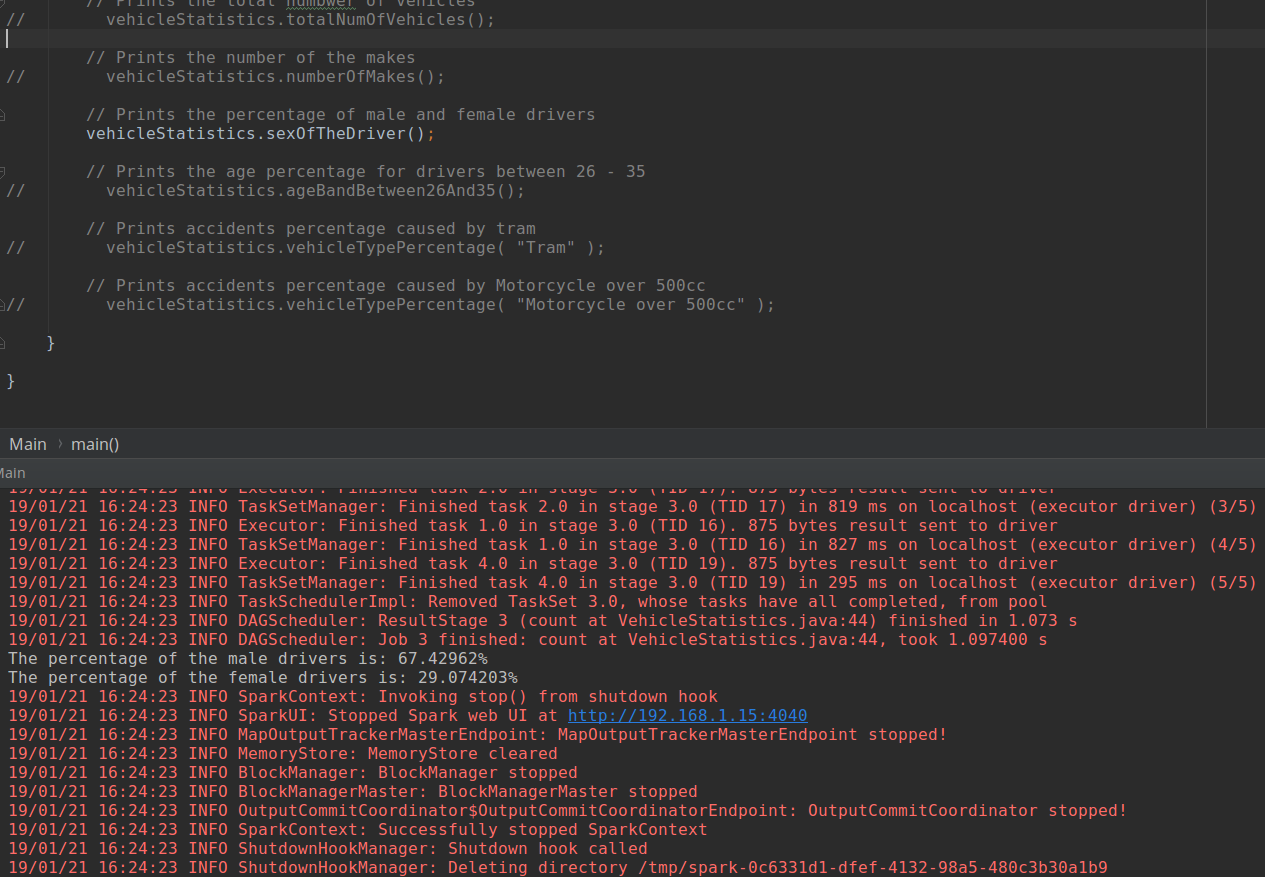
Εικόνα 2.5

1. Στην εικόνα 2.6 μπορούμε να δούμε τον πλήθος το εταιριών αυτοκινήτων. Όπως φαίνεται και στην εικόνα είναι 536.



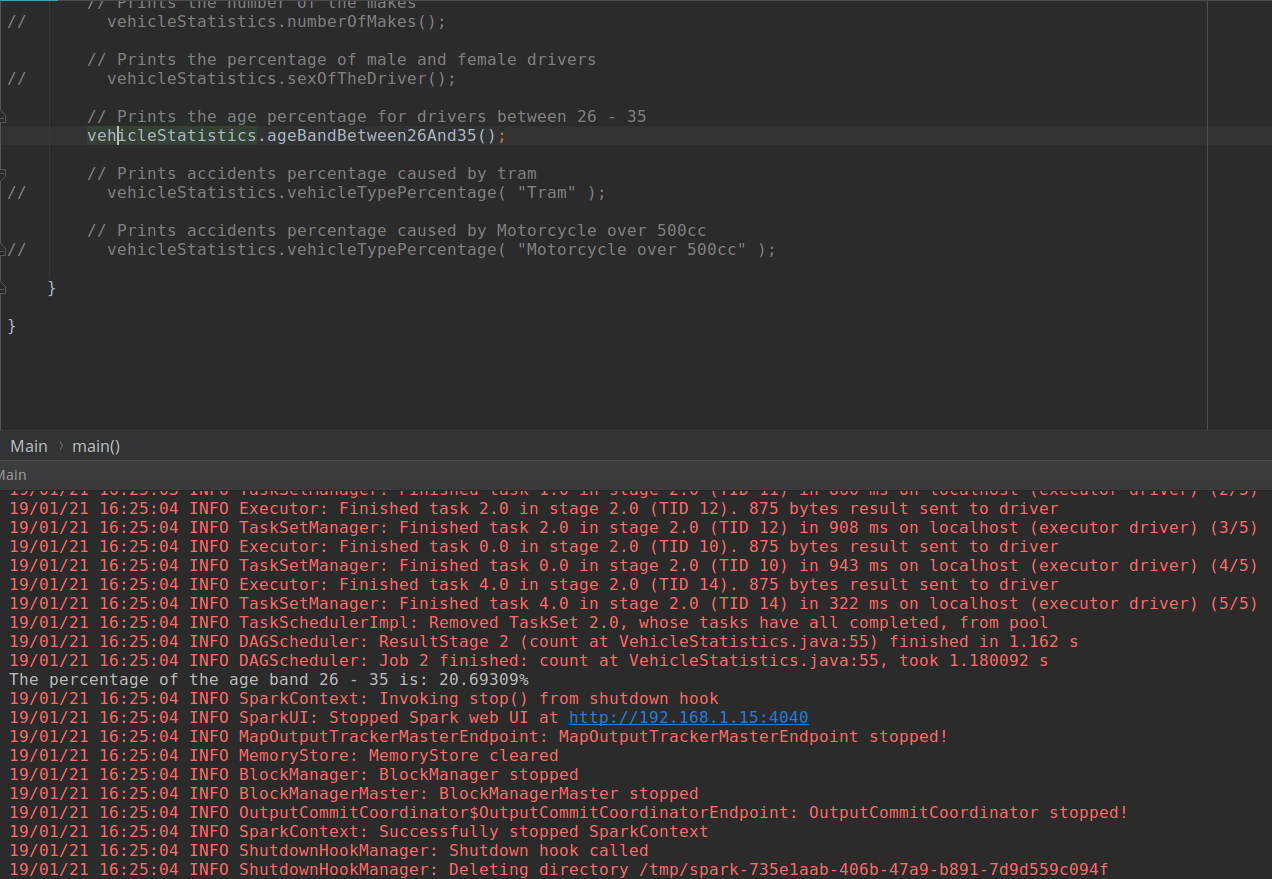
Εικόνα 2.6

1. Στην εικόνα 2.7 μπορούμε να δούμε τα ποσοστά επι της εκατό των δυο φύλων που ενεπλάκησαν σε κάποιο ατύχημα. Για τους άντρες έχουμε 67,42962% και για τις γυναίκες 29.074203%.



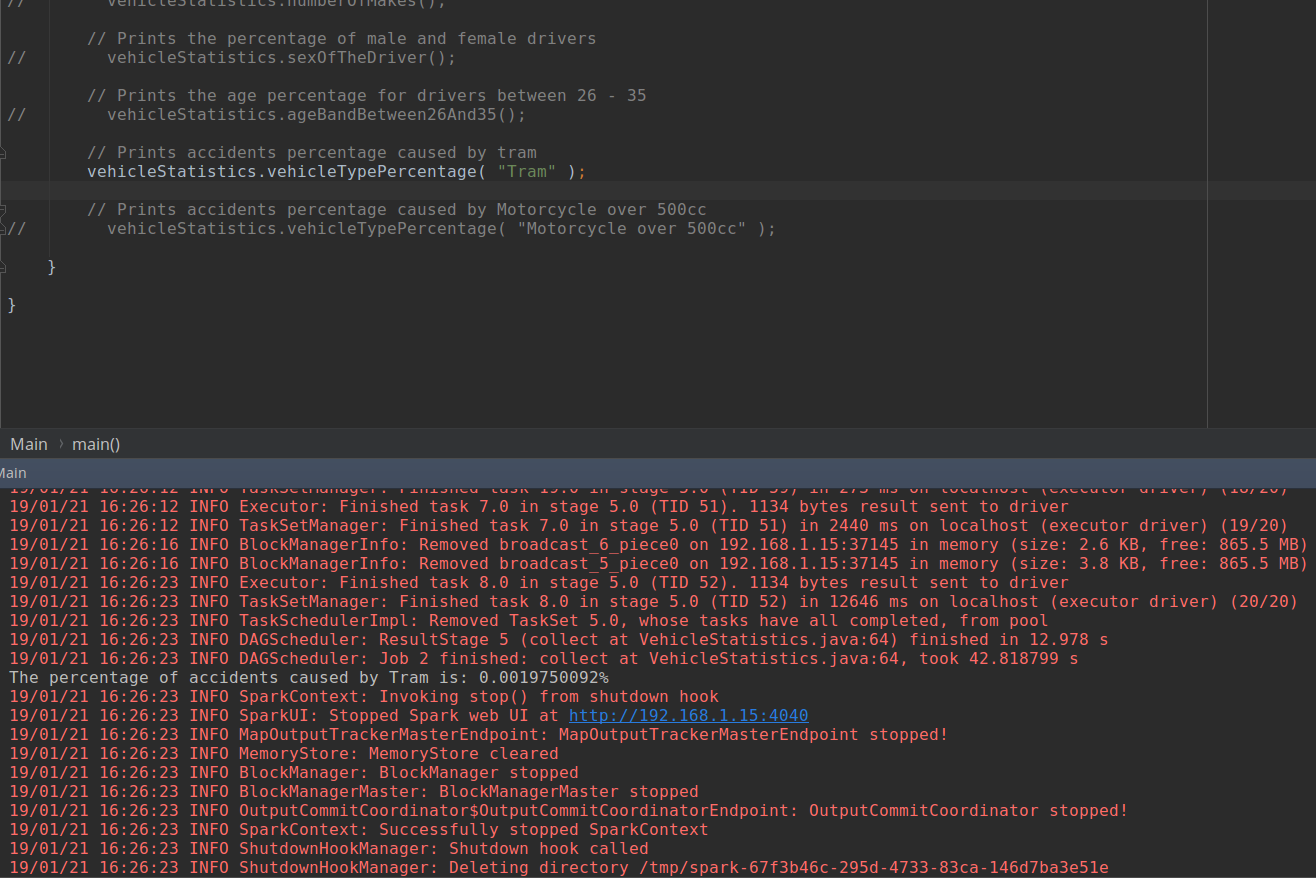
Εικόνα 2.7

1. Στην εικόνα 2.8 μπορούμε να δούμε τα ποσοστά επι της εκατό των εμπλεκομένων σε ατύχημα που ήταν μεταξύ 26 και 35 ετών.



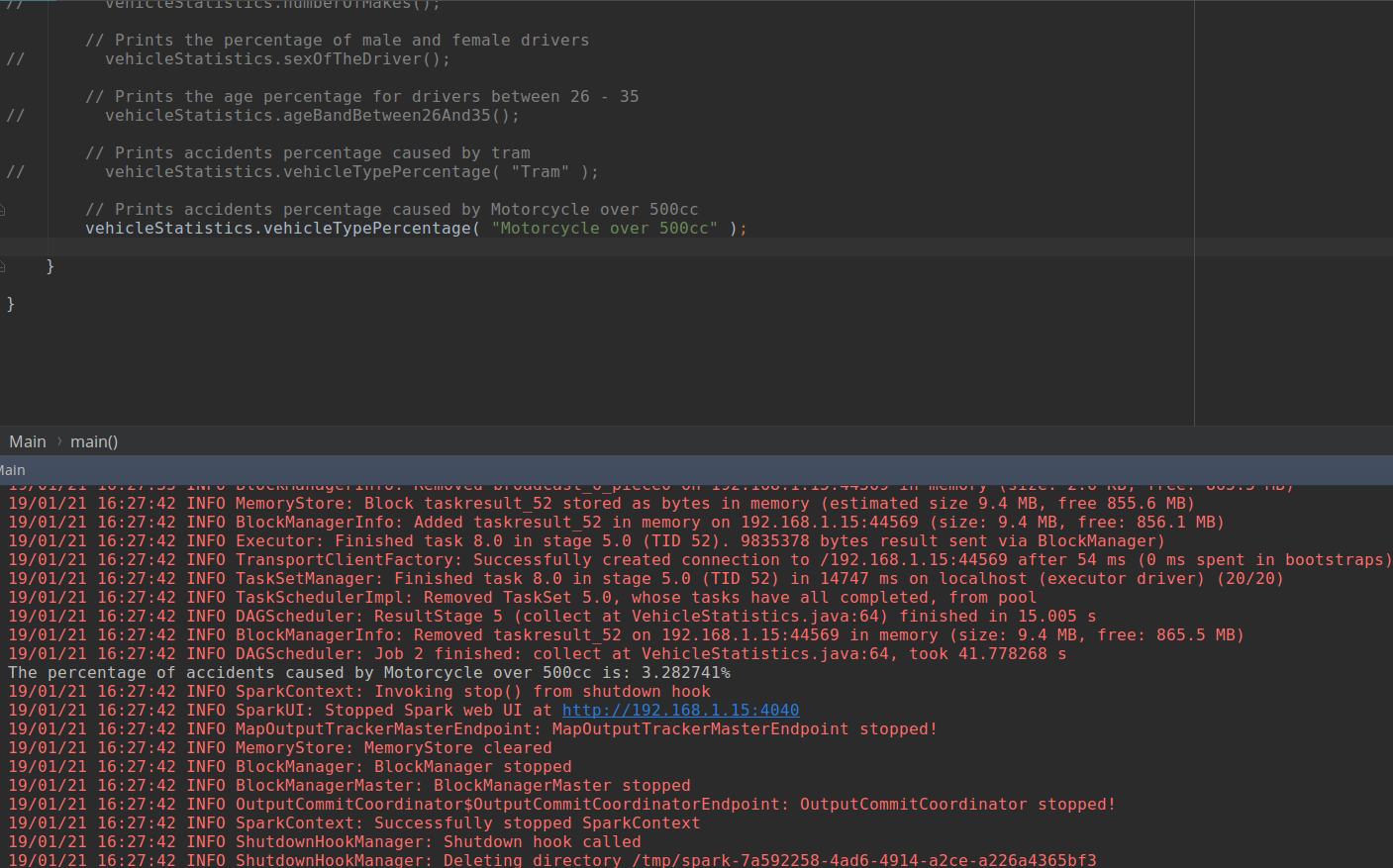
Εικόνα 2.8

1. Στην εικόνα 2.9 μπορούμε να δούμε τα ποσοστά επι της εκατό των ατυχημάτων στα οποία το όχημα ήταν τραμ.



Εικόνα 2.9

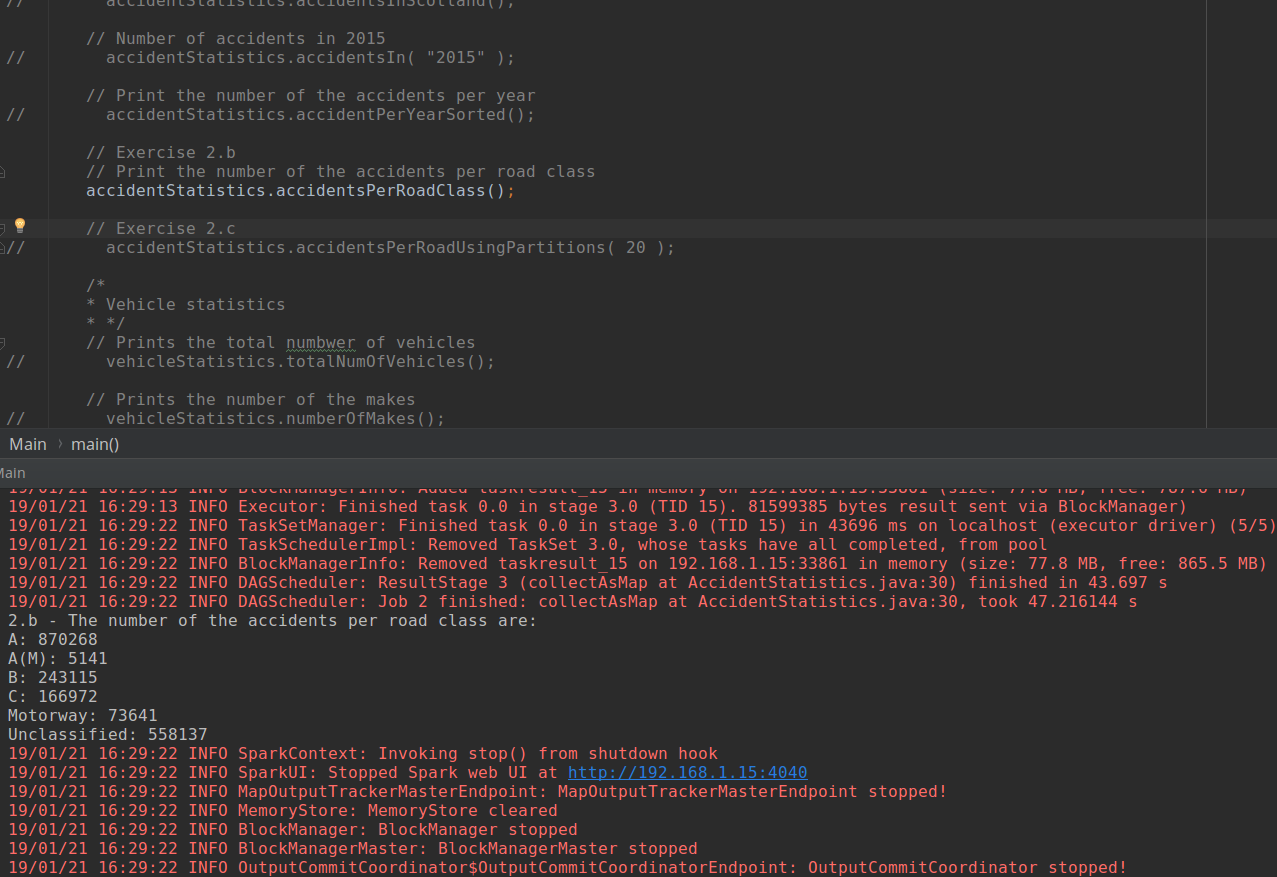
1. Στην εικόνα 2.10 μπορούμε να δούμε τα ποσοστά επι της εκατό των ατυχημάτων στα οποία το όχημα ήταν μοτοσυκλέτα άνω των 500 κυβικών.



Εικόνα 2.10

## Ερώτημα Β

Το υποερωτημα που επιλέξαμε ,από το μέρος 1, ώστε να το προσαρμόσουμε στο Spark είναι το 1.a.i. Όπως φαίνεται στην εικόνα 2.11 εκτελώντας τη μέθοδο accidentsPerRoadClass παίρνουμε το αντίστοιχο αποτέλεσμα με το ερώτημα προς μετατροπή.

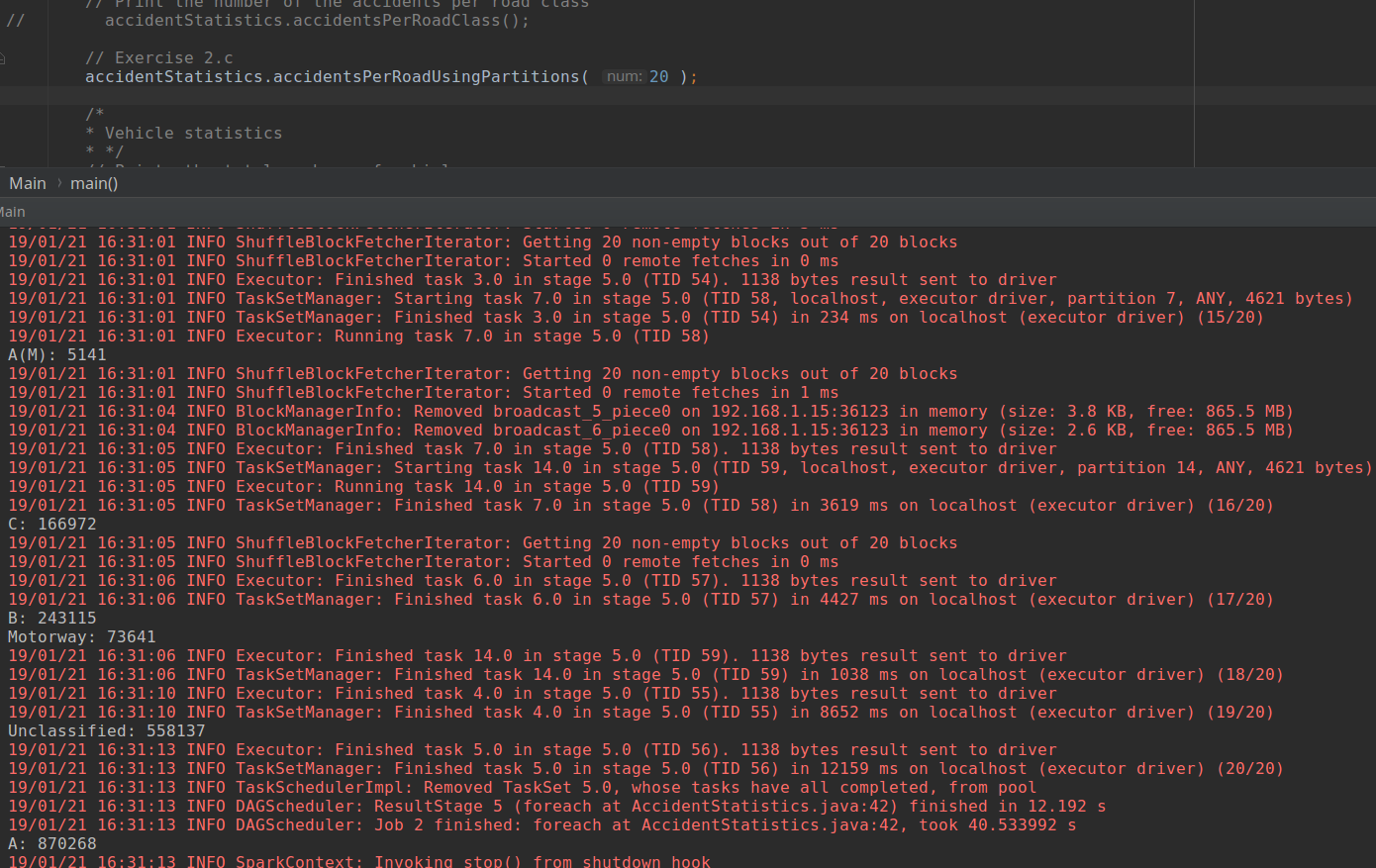


Εικόνα 2.11

## Ερώτημα C

Για το τρίτο και τελευταίο ερώτημα επιλέξαμε να κάνουμε hash partitioning. Στο παράδειγμα που παρουσιάζουμε στην εικόνα 2.12 καλούμε τη μέθοδο accidentPerRoadUsingPartitions και ως όρισμα δίνουμε τον αριθμό των partition που θέλουμε. Στην περίπτωση μας, 20.

Μια παρατήρηση που μπορούμε να κάνομε είναι ότι τα αποτελέσματα εμφανίζονται ασύγχρονα σε σχέση με το ερώτημα B, δηλαδή όποτε είναι έτοιμό κάποιο αποτέλεσμά τυπώνεται στην οθόνη.



Εικόνα 2.12